



Instituto Politécnico de Tomar

**Escola Superior de Tecnologia de Tomar**

**David Rafael Marmelo Carrilho**

# **Desenvolvimento e manutenção para sistemas SAP ERP**

Relatório de Estágio

Orientado por:

Prof. Doutor Manuel Fernando Martins de Barros, Instituto Politécnico de Tomar  
Mestre Carlos Daniel da Silva Reis Pires, Softinsa, Lda.

Relatório de Estágio  
apresentado ao Instituto Politécnico de Tomar  
para cumprimento dos requisitos necessários  
à obtenção do grau de Mestre em Engenharia Informática  
- Internet das Coisas





*“The best way to predict the future is to invent it.”*

Alan kay.



## Resumo

---

Os negócios atuais exigem tecnologias inovadoras que preencham as lacunas entre os processos de negócios e a sua interação com as pessoas de forma a garantir maior produtividade, eficiência e rentabilidade. As novas organizações de trabalho possuem vários departamentos e equipas técnicas com competências para executar com sucesso este tipo de interação. Dada a complexidade das operações e a exigência requerida na fiabilidade e qualidade dos dados de uma empresa, é de todo aconselhável a utilização de um sistema que dê suporte e coordene todas as informações e comunicações dentro de uma organização, tal como os modernos sistemas ERP. Um sistema ERP é um sistema moderno de planeamento e controlo integrado de gestão de toda a cadeia produtiva de uma empresa.

Durante o estágio na Softinsa, o estagiário aplicou em prática o conhecimento adquirido tanto na Licenciatura de Engenharia Informática, como no Mestrado em Engenharia Informática - Internet das Coisas. A nível técnico, este estágio permitiu a aprendizagem do *software* SAP ERP, a linguagem de programação ABAP e do add-on Neptune envolvendo JavaScript, CSS e HTML. Todos estes projetos foram desenvolvidos com base em metodologias de desenvolvimento SCRUM e Waterfall.

No presente relatório está descrito o desenvolvimento de um projeto e algumas tarefas de manutenção em sistemas SAP realizadas pelo estagiário. O desenvolvimento de um projeto sobre o modulo de Plant Maintenance para computador e dispositivos móveis usando o add-on Neptune, e a implementação de notas que consiste na instalação de um conjunto de correções para melhorar um sistema.

Através da realização deste estágio, foram também adquiridos conhecimentos ao nível da programação e a nível profissional. O estágio permitiu um contacto direto com o mercado de trabalho e permitiu aplicar em prática muitos conceitos teóricos adquiridos durante a licenciatura e o mestrado, captando assim um conjunto de práticas do mundo empresarial e metodologias de trabalho que serviram de exemplo para o futuro profissional. No futuro, o estagiário pretende continuar a trabalhar e a explorar as soluções SAP, aprofundar os conhecimentos da programação ABAP e Neptune e a utilizar as tecnologias SAP Leonardo.

**Palavras-chave:** SAP, ERP, ABAP, Neptune, SAP Leonardo.



# Abstract

---

Today's businesses require innovative technologies that bridge the gaps between business processes and their interaction with people to ensure greater productivity, efficiency and profitability. The new work organizations have various departments and technical teams with skills to successfully perform this type of interaction. Given the complexity of operations and the required reliability and data quality requirements of a company, it is highly advisable to use a system that supports and coordinates all information and communications within an organization, such as modern ERP systems. An ERP system is a modern integrated planning and management control system for the entire production chain of a company.

During the internship at Softinsa, the intern applied in practice the knowledge acquired both in Computer Engineering Degree, as well as in the Master in Computer Engineering - Internet of Things. At the technical level, this internship has enabled learning of SAP ERP software, the ABAP programming language and the Neptune add-on involving JavaScript, CSS and HTML. All of these projects were developed based on SCRUM and Waterfall development methodologies.

This report describes the development of a project and some maintenance tasks on SAP systems performed by the intern. Developing a Plant Maintenance module project for computer and mobile devices using the Neptune add-on, and implementing notes consisting of installing a set of patches to improve a system.

Through this internship, knowledge was also acquired at the programming and professional level. The internship allowed a direct contact with the labor market and allowed to apply in practice many theoretical concepts acquired during the undergraduate and masters degree, thus capturing a set of business practices and work methodologies that served as an example for the future professional. In the future, the trainee intends to continue working and exploring SAP solutions, deepening knowledge of ABAP and Neptune programming, and using SAP Leonardo technologies.

**Keywords:** SAP, ERP, ABAP, Neptune, SAP Leonardo.





## Agradecimentos

---

A realização deste relatório de estágio de Mestrado contou com vários incentivos e apoios de muitas pessoas que sem elas não teria conseguido tornar este relatório uma realidade.

Ao Professor Doutor Manuel Fernando Martins de Barros, pela preocupação e disponibilidade em querer ajudar o desenvolvimento deste relatório.

Ao meu orientador de estágio, Mestre Carlos Daniel da Silva Reis Pires, pelos conhecimentos transmitidos ao longo do estágio, pela ajuda a superar alguns desafios que foram aparecendo ao longo do estágio.

À minha namorada, que me apoia, ajuda, motiva a querer ser mais e melhor e pela paciência que teve durante todo o meu percurso académico e do mestrado.

A todos os meus amigos que me motivaram a tirar o mestrado, porque se não fosse agora, já não era.

E por fim, e sempre importante, à minha família, por toda a paciência, preocupação e ajuda que me deram durante esta etapa da minha vida.

A todos um muito obrigado!



# Índice

Resumo.....	vi
Abstract .....	viii
Agradecimentos.....	x
Índice .....	xii
Índice de Figuras .....	xv
Índice de Tabelas.....	xvii
Lista de Abreviaturas e Siglas .....	xix
Capítulo 1   Introdução .....	1
1.1 Contextualização .....	1
1.2 Motivação e Objetivos .....	2
1.3 Relevância do Estágio .....	3
1.4 Organização do Relatório.....	4
Capítulo 2   Caracterização do Meio Envolvente .....	5
2.1 Empresa.....	5
2.2 Estrutura Organizacional.....	6
2.3 Departamento de Global Business Services (GBS) .....	7
2.4 Consultor de GBS.....	7
Capítulo 3   Sistemas de Informação .....	9
3.1 Sistemas de Informação .....	9
3.2 Enterprise Resource Planning (ERP) .....	9
3.3 Systems Applications and Products (SAP) .....	10
3.3.1 Módulos SAP .....	11
3.3.2 Fluxo de Trabalho entre Equipas .....	12
3.4 Advanced Business Application Programming (ABAP) .....	14
3.5 SAP Fiori.....	15

3.6 Neptune.....	16
3.7 SAP Leonardo .....	20
3.7.1 Soluções SAP Leonardo .....	20
3.8 SAP HANA .....	22
Capítulo 4   Metodologias de projeto .....	23
4.1 Breve introdução.....	23
4.1 Metodologias Ágeis e Tradicionais .....	24
4.2 Metodologia SCRUM.....	25
4.3 Metodologia <i>Waterfall</i> .....	26
4.3 Comparação de metodologias.....	28
Capítulo 5   Apresentação e descrição das atividades realizadas .....	29
5.1 Atividades Realizadas .....	29
5.2 Caso 1   Projeto.....	29
5.2.1 Projeto.....	29
5.2.2 Análise de Requisitos para aplicações de demonstração.....	30
5.2.2.1 Aplicação de equipamentos .....	32
5.2.2.2 Aplicação de locais de instalação .....	36
5.2.2.3 Aplicação de ordens de trabalho só com detalhes .....	37
5.2.2.4 Aplicação de notificações com funcionalidades de PM .....	38
5.2.2.5 Aplicação de ordens de trabalho com funcionalidades de PM.....	43
5.2.3 Análise de Requisitos para aplicações vendidas a um cliente .....	49
5.2.3.1 Aplicação de notificações com funcionalidades PM.....	50
5.2.3.2 Aplicação de ordens de trabalho com funcionalidades PM.....	53
5.2.4 Transações Principais .....	57
5.2.5 Conclusão .....	58
5.3 Caso 2   Implementação de Notas.....	59

5.3.1 Problema.....	59
5.3.2 Análise de Requisitos .....	60
5.3.3 Resolução .....	61
5.3.4 Conclusão .....	64
Capítulo 6   Conclusões .....	65
Referências Bibliográficas .....	69

## Índice de Figuras

Figura 1- Organograma Softinsa [5].....	6
Figura 2 - Lógica de um sistema ERP [18]. ....	11
Figura 3 - Fluxo de trabalho entre equipas. ....	13
Figura 4 - Arquitetura do sistema ABAP [24].....	14
Figura 5 - SAP Fiori [25].....	15
Figura 6 – Launchpad do Fiori [25].....	16
Figura 7 - Ciclo de vida de uma aplicação em Neptune [28]. ....	17
Figura 8 - Ambiente de desenvolvimento em Neptune. ....	18
Figura 9 - Painel de performance de aplicações Neptune. ....	18
Figura 10 - Launchpad do Neptune [30]. ....	19
Figura 11 - Fluxo de dados simples numa aplicação Neptune. ....	19
Figura 12 - Tecnologias que constituem o SAP Leonardo [33]. ....	21
Figura 13 - Funcionamento do SAP HANA [37].....	22
Figura 14 - Breve visão das metodologias Ágeis [42].....	24
Figura 15 - Fluxo de trabalho utilizando a metodologia SCRUM [45].....	26
Figura 16 - Fluxo de trabalho utilizando a metodologia <i>Waterfall</i> [47].....	27
Figura 17 - <i>Tiles</i> das aplicações genéricas no <i>launchpad</i> na máquina <i>sandbox</i> da Softinsa. .....	31
Figura 18 - Dependência da aplicação de listagem de ordens com CRUD.....	31
Figura 19 - Aplicação genérica de listagem e detalhes de equipamentos. ....	33
Figura 20 - Aplicação genérica de listagem de equipamentos sem detalhes.....	33
Figura 21 - Aplicação genérica de listagem de equipamentos sem detalhes modo <i>mobile</i> (tablet) com botão de leitura de código de barras.....	34
Figura 22 - Câmara em modo de leitura de código de barras.....	35
Figura 23 - Diagrama de casos de uso para aplicação de listagem de equipamentos.....	35
Figura 24 - Aplicação genérica de listagem de locais de instalação com detalhes. ....	36
Figura 25 - Aplicação genérica de listagem de ordens de trabalho com detalhes, mas sem funcionalidades de PM. ....	37
Figura 26 - Diagrama de casos de uso para aplicação de listagem de ordens de trabalho com detalhes, mas sem funcionalidades de PM. ....	38

Figura 27 - Aplicação genérica de listagem de documentos de notificações com funcionalidades de PM. ....	40
Figura 28 - Aplicação genérica de CRUD de uma notificação em modo editar. ....	41
Figura 29 - Diagrama de casos de uso para aplicação de notificações genérica com funcionalidades de PM. ....	42
Figura 30 - Aplicação genérica de listagem de ordens de trabalho com funcionalidades de PM. ....	44
Figura 31 - Aplicação genérica de CRUD de uma ordem de trabalho em modo editar. ....	45
Figura 32 - Aplicação genérica de CRUD de uma ordem de trabalho em modo editar listando operações de trabalho. ....	46
Figura 33 - Aplicação genérica de confirmações de operações e componentes. ....	47
Figura 34 - Diagrama de casos de uso para aplicação genérica de ordens de trabalho com funcionalidades de PM. ....	48
Figura 35 - <i>Tiles</i> das aplicações compradas pelo cliente no <i>launchpad</i> da sua máquina. ....	49
Figura 36 - Aplicação de listagem de documentos de notificações com funcionalidades de PM adaptada ao cliente A. ....	50
Figura 37 - Aplicação de CRUD de uma notificação em modo editar. ....	51
Figura 38 - Diagrama de casos de uso para aplicação para cliente de notificações com funcionalidades de PM. ....	52
Figura 39 - Aplicação de listagem de documentos de ordens de trabalho com funcionalidades de PM adaptada ao cliente A. ....	54
Figura 40 - Aplicação de CRUD de uma ordem de trabalho em modo editar. ....	55
Figura 41 - Diagrama de casos de uso para aplicação para cliente de ordens de trabalho com funcionalidades de PM. ....	56
Figura 42 – Tabela de parametrização a utilizadores no cliente ....	59
Figura 43 – Módulos, <i>releases</i> e <i>suppot package</i> instalados na <i>sandbox</i> da Softinsa. ....	60
Figura 44 - Módulo, <i>release</i> e <i>suppot package</i> em que é para implementar a nota. ....	61
Figura 45 - Indicação de número de passos manuais da nota. ....	62
Figura 46 - Detalhe da nota antes de ser implementada. ....	62
Figura 47 - Estado do objeto a ser alterado pela nota. ....	63
Figura 48 - Lista de objetos que têm de ser ativados depois da implementação da nota. ....	63
Figura 49 - Detalhe da nota depois de ser implementada. ....	63



## **Índice de Tabelas**

Tabela 1 - Módulos de SAP ERP [19].....	12
Tabela 2 - Diferenças entre metodologias tradicionais e as metodologias ágeis [48] [50]...	28
Tabela 3 - Principais transações utilizadas .....	57



## **Lista de Abreviaturas e Siglas**

ABAP – *Advanced Business Application Programming*

CENIT – Centro de Inovação Tecnológica

CRUD – *Create, Read, Update, Delete*

ERP – *Enterprise Resource Planning*

GBS – *Global Business Services*

GTS – *Global Technology Services*

IT – *Information Technology*

PM – *Plant Maintenance*

SAP – *Systems, Applications and Products in Data Processing*

SI – Sistemas de Informação



## Capítulo 1 | Introdução

### 1.1 Contextualização

Nos dias de hoje, a sociedade, as instituições e as empresas tem colocado em ênfase cada vez maior, na recolha e interpretação precisa dos dados. Os dados são considerados a base da pirâmide ideal de uma decisão pragmática [1], isto é, uma decisão que envolve um diálogo entre especialistas e decisores. A primeira etapa de um processo de decisão pragmática é a recolha de dados relativos ao assunto sobre o qual se pretende tomar decisões. Dados sem serem tratados, de nada informam e por isso, torna difícil uma tomada de decisão objetiva e fundamentada. É através da análise e tratamentos de dados que podemos recolher informação e ajudar à melhor tomada de decisão.

Uma das empresas que impulsionou muito a área da informação foi a SAP, através da criação de soluções ERP (*Enterprise Resource Planning*) para os seus clientes. O uso de soluções da SAP não só permite às empresas a consulta de informação em tempo real como também permite uma gestão estratégica muito mais eficiente, facilitando a tomada de decisão e prevenindo situações menos positivas para a organização.

SAP é um *software* de *Enterprise Resource Planning* (ERP) da empresa SAP AG, Alemanha. As soluções SAP são usadas em empresas de grande ou médio porte para integrar os processos em várias funções. Cada função é tratada pelo respetivo módulo no *software* SAP. Por exemplo, a SAP possui módulos chamados *Sales & Distribution* (SD), *Advanced Planning and Optimization* (APO), *Quality Management* (QM), *Purchasing* (MM-PUR) *Warehouse Management* (WM), *Production Planning* (PP), *Finance* (FI), *Plant Maintenance* (PM) e muitos outros.

A SAP foi fundada em 1972 por cinco empresários na Alemanha que tinham uma visão para o potencial de negócios da tecnologia. Começando com um cliente e alguns funcionários, a SAP estabeleceu um caminho que não apenas transformou o mundo da tecnologia da informação, mas também alteraria para sempre a forma como as empresas fazem negócios [2].

Para o desenvolvimento das soluções ERP da SAP, é utilizada a linguagem de programação ABAP. Devido à necessidade dos utilizadores acederem à informação em

qualquer lugar e a qualquer altura, foi desenvolvida uma ferramenta de desenvolvimento de aplicações SAP para dispositivos móveis: a Neptune. Esta ferramenta permite que seja possível a criação de aplicações móveis, através da linguagem ABAP e de JavaScript.

## **1.2 Motivação e Objetivos**

O estágio é um instrumento que permite estimular e reforçar as competências de um estudante, tornando o processo de ensino numa experiência enriquecedora. A motivação que me levou a realizar um estágio na empresa Softinsa, em vez de um projeto foi a procura de experiência profissional. A inserção de um estudante no mercado de trabalho, por meio de um estágio, permite colocar em prática os conceitos teóricos adquiridos durante a formação académica e por outro lado, assimilar um conjunto de metodologias e práticas do mundo empresarial que servirá de exemplo para o futuro profissional.

O que me motivou a fazer um estágio em consultadoria SAP ABAP e Neptune foi o desafio de aprender uma área nova da engenharia informática que acaba por ser um pouco esquecida. Um bom sistema ERP é o que cria as expectativas do mundo empresarial de hoje, com acesso a toda a informação disponível a qualquer momento para facilitar o trabalho de qualquer pessoa. Este é o motivo que me leva a querer procurar um novo desafio numa área pouco conhecida por estudantes acabados de se formar.

Este estágio proporcionou uma oportunidade de trabalho na área de consultadoria SAP ABAP e Neptune. O objetivo principal é desenvolver e apresentar soluções aos clientes da Softinsa, conforme os requisitos de cada empresa. As soluções propostas podem ser melhorias, correções, ou até o desenvolvimento de projetos específicos desenvolvidos ao nível da programação ABAP e Neptune sobre o sistema ERP da SAP. Um bom sistema ERP deve ser moldado às necessidades do cliente, sendo capaz de processar os mais variados tipos de dados, recursos e processos como faturamento, balancetes, controle de gastos, fluxos de caixa, inventário, stock, controle de ativos e equipamentos, tempo de processamento, etc.

Em resumo, o objetivo geral do estágio é a execução de tarefas a nível da programação, de forma a que seja possível criar e manter aplicações SAP e aplicações Neptune sobre SAP.

Os objetivos específicos do estágio são:

- Aprendizagem de programação em SAP, com recurso à linguagem ABAP;
- Criação de competências em Neptune e em linguagem de programação Javascript;
- Desenvolvimento de programas em ABAP;
- Desenvolvimento de aplicações móveis em Neptune;
- Manutenção de programas em ABAP e Neptune;
- Consolidação de conhecimentos adquiridos no mestrado.

### **1.3 Relevância do Estágio**

Durante o estágio, o estagiário irá aplicar em prática o conhecimento adquirido tanto na Licenciatura de Engenharia Informática, como no Mestrado em Engenharia Informática - Internet da Coisas.

A nível técnico, este estágio permitirá a aprendizagem da linguagem de programação ABAP e do *addon* Neptune. O estagiário irá desenvolver competências de um Consultor SAP ABAP/Neptune e realizará tarefas de criar, corrigir ou melhorar programas ou aplicações em SAP e Neptune.

Ao longo do estágio, serão aplicados conhecimentos aprendidos no mestrado nas disciplinas de Engenharia de Software, Análise e processamento de grandes volumes de dados e Gestão de projetos em Engenharia Informática. Também serão consolidados conhecimentos de base de dados e de JavaScript.

O estágio numa empresa com a dimensão e prestígio da SoftInsa, permitirá o contacto privilegiado com pessoas com uma vasta experiência nesta área e com a capacidade de transmitir competências e conhecimentos que serão importantes para o desenvolvimento profissional do estagiário.

## 1.4 Organização do Relatório

No capítulo 1, é feita a introdução do relatório, que contextualiza o tema do estágio e apresenta a motivação, objetivos, e a relevância do estágio para o aluno, assim como, é apresentada a organização do relatório.

No capítulo 2, é feita uma caracterização do meio envolvente, onde se descreve a empresa em que o estágio é feito, a sua estrutura organizacional, o departamento de *Global Business Services* e qual o papel de um consultor SAP ABAP e Neptune na empresa.

O capítulo 3 descreve os sistemas de informação utilizados durante o estágio e também outras soluções SAP. Faz-se uma introdução aos sistemas de informação, em que consiste um ERP, o que é a SAP, como funciona e quais as soluções que fornecem aos seus clientes. Também é descrito as tecnologias utilizadas ao longo do estágio e qual a sua importância.

No capítulo 4, são apresentadas as metodologias de trabalho em desenvolvimentos de projetos e do *software* utilizado no desenvolvimento de soluções SAP ABAP e Neptune. É feita uma introdução às metodologias ágeis e tradicionais, metodologia SCRUM, metodologias *Waterfall* e no final é feita uma avaliação comparativa destas metodologias.

No capítulo 5 é feita uma descrição detalhada das atividades realizadas ao longo do estágio. Cada sub-capítulo é dividido genericamente por caso, problema, análise de requisitos, resolução, transações principais e conclusão. O primeiro caso descreve o projeto de *Plant Maintenance* em que o estágio se incidiu. O segundo caso é descrito uma fase de implementação de notas que consiste num conjunto de correções e melhorias no sistema.

No capítulo 6 é apresentada a conclusão e as reflexões e considerações finais que resultaram do estágio.



## **Capítulo 2 | Caracterização do Meio Envolvente**

### **2.1 Empresa**

A Softinsa é uma empresa do grupo IBM e da Viewnext.

No ano 1994 foi criada em Portugal a empresa CGI Portugal – Compagnie Generale D’Informatique Portugal. Esta é uma empresa francesa pertencente ao Grupo IBM, com delegações em diversos países do Mundo. A CGI Portugal dedicava-se exclusivamente à comercialização, implementação, desenvolvimento e manutenção de uma Plataforma para Gestão de Recursos Humanos e Imobilizado [5].

No dia 1 de abril de 1998 a INSA, S.A. – Ingenieria de Software Avanzado (empresa espanhola constituída em 1991 e também pertencente à IBM Espanha), atual Viewnext, transforma a CGI Portugal na sua sucursal portuguesa e abre um escritório em Lisboa, alargando assim a sua abrangência na Península Ibérica [5].

Com esta transformação, a INSA começou a oferecer serviços na área das tecnologias da informação que iam desde a área de sistemas passando pelo desenvolvimento aplicacional e à implementação do ERP SAP [5].

No início de agosto de 2007 a Sucursal Portuguesa da Viewnext transforma-se numa empresa independente de nome Softinsa – Engenharia de Software Avançado Lda., detida pela Companhia IBM Portuguesa e pela Viewnext, dando continuidade ao trabalho desenvolvido até então, mas reforçando as sinergias locais com a IBM Portugal [5].

Com 21 anos de história e experiência no mercado português, conta atualmente com uma equipa de mais de 800 profissionais em tecnologias de informação e com uma cobertura ibérica de escritórios e Centros de Inovação Tecnológica (CENIT’s) [5]. A rede Softinsa inclui delegações em Lisboa e Centros de Inovação Tecnológica em Tomar (desde 2013) [3] e Viseu (desde 2016) [4].

Os Centros de Inovação Tecnológicas são especializados na prestação de serviços e na gestão e desenvolvimento de aplicações, com capacidade regional e global, e integram a rede internacional dos Centros de Inovação da IBM. Estes centros colocam à disposição dos seus clientes as infraestruturas, as ferramentas e os profissionais capazes de responder às

mais diversas necessidades na área das tecnologias de informação, oferecendo serviços personalizados, como a gestão remota de infraestruturas e aplicações ou processos [7].

A Softinsa é especializada nas áreas de *Application Management Services, Analytics, Cognitive, Mobile, Asset Management, Human Capital Solutions, SAP, IT Managed Services, Data Center Facilities, Mobility e Cloud Managed Services* [5].

Para manter um padrão de qualidade e excelência dos serviços com os seus clientes, a SoftInsa aposta nas certificações da qualidade dos seus serviços e soluções. A Softinsa é certificada, desde novembro de 2002, de acordo a norma de referência ISO 9001:2015, pelo Sistema de Gestão da Qualidade da Bureau Veritas Quality International, tendo o Centro de Inovação Tecnológica de Tomar obtido, em novembro de 2015 [5].

## 2.2 Estrutura Organizacional

A Softinsa está organizada hierarquicamente da seguinte forma (Fig. 1):

- Diretor Geral - Sérgio Pereira
- Diretor de Estratégia - Henrique Mourisca
- Diretor de Desenvolvimento do Negócio - José Caratão
- Diretor do Departamento de GBS (*Global Business Services*) - Nuno Alves
- Diretor do Departamento de GTS (*Global Technology Services*) - Francisco Caeiro
- Diretor do Departamento Financeiro - Paulo Gervásio
- Diretora do Departamento de Recursos Humanos - Raquel Jerónimo



Figura 1- Organograma Softinsa [5].

Em relação à componente de SAP, a mesma está definida e organizada no departamento de GBS, o Henrique Coelho é o seu gestor. Em relação à área de ABAP, o gestor é o Carlos Pires. Também é a pessoa que coordena e distribui as tarefas a realizar pelos consultores da equipa de ABAP nos diversos centros da Softinsa (escritório em Lisboa e CENIT's de Tomar e Viseu). Por fim estão os consultores SAP ABAP/Neptune na qual o estagiário se insere.

## **2.3 Departamento de Global Business Services (GBS)**

A equipa de *Global Business Services* da Softinsa permite um suporte constante aos seus clientes, nos mais diversos setores de atividade, ajudando os mesmos a atingir os seus objetivos. A equipa de GBS fornece o seu suporte aos seus clientes, independentemente da sua localização, fazendo com que seja possível fazer uma melhor gestão dos recursos que dispõe [6].

Esta equipa foi criada com o objetivo de oferecer os recursos necessários que atendam aos requisitos dos seus clientes que baseiam o seu ERP em sistemas SAP. Atualmente, a equipa de GBS da Softinsa é constituída por mais de 350 consultores, programadores e analistas, especializados no desenvolvimento, implementação e gestão de soluções tecnológicas [6].

## **2.4 Consultor de GBS**

Sendo a Softinsa uma empresa parceira da IBM, segue as suas normas organizacionais, dividindo-se em dois ramos de serviços profissionais, *Global Technology Services* (GTS) e *Global Business Services* (GBS) [8].

- GTS lida com serviços de tecnologias de informação, suporte e implementação de infraestrutura principalmente relacionadas a serviços.
- GBS lida com consultoria estratégica, integração de sistemas, gestão de aplicações, entre outros.

A equipa de GBS é constituída por consultores de negócios, estratégia e tecnologia que ajudam os clientes a visualizar o seu futuro como empresas cognitivas, todas suportadas

pelo modelo de talentos mais sofisticado do setor. Com alcance global, metodologias focadas em resultados e profunda experiência no setor, o GBS capacita os clientes a reinventar digitalmente seus negócios e obter vantagem competitiva [8].

Um consultor SAP faz parte da equipa de GBS e pode se especializar em diversas áreas, entre elas o consultor de negócios (funcional) e o programador, oferecendo auxílio no desenvolvimento e na implantação de módulos SAP, visando o lucro e desenvolvimento das suas empresas [9] [10].

Após o consultor SAP implementar a solução proposta pelo cliente, ainda poderá auxiliar o cliente, com o suporte local ou remoto, efetuando desenvolvimento de novos processos, personalizações, layouts, relatórios e estando constantemente em contato com o cliente [9] [10].

## Capítulo 3 | Sistemas de Informação

Este capítulo faz uma breve introdução dos sistemas de informação utilizados durante o estágio. Faz-se uma descrição dos sistemas de informação e do sistema ERP, do *software* SAP e quais as soluções fornecidas por esta aplicação aos clientes finais. Também são descritas as tecnologias utilizadas ao longo do estágio e qual a sua importância.

### 3.1 Sistemas de Informação

Um Sistema de Informação (SI) é um sistema constituído por pessoas, procedimentos e equipamentos que recolhe, processa, armazena, e distribui informações com objetivos específicos [11].

Como qualquer outro Sistema, um Sistema de Informação é composto por *inputs* (dados, instruções) e *outputs* (relatórios, cálculos). O SI processa os *inputs* e produz *outputs* que são disponibilizados ao utilizador final ou a outros sistemas. Costuma também ser incluído um mecanismo de *feedback* que controla a operação [11].

A sua utilização traz numerosas vantagens, como a otimização do processo de tomada de decisão e eliminação de redundância de atividades, também permite aceder, compreender e responder à informação de forma rápida e eficaz. No entanto, a utilização de um SI não torna só por si uma empresa verdadeiramente integrada e tem altos custos que podem não justificar o custo/benefício [12] [13] [14].

### 3.2 Enterprise Resource Planning (ERP)

ERP (*Enterprise Resource Planning*), Planeamento de Recursos Empresariais, é um *software* de gestão de processos de negócio que engloba vários módulos e que suportam diversas áreas funcionais, sendo os mais comuns, as atividades de finanças, a cadeia de fornecimento, as operações, os relatórios, o fabrico e os recursos humanos. A maior parte das empresas possuem um sistema de finanças e operações, mas não é capaz de acompanhar o crescimento do negócio. Por esse motivo, as empresas precisam de mudar e os sistemas que possuem devem acompanhar o respetivo crescimento. [20]

A grande vantagem de um sistema ERP é sua capacidade de integração das informações do negócio de forma ampla e abrangente, evitando dados duplicados e disponibilizando informações de uma área da empresa para todas as outras, de forma ágil e em tempo real.

Os sistemas ERP têm um impacto significativo no mundo empresarial e apresentam várias vantagens, tais como [21]:

- Auxílio na implantação das melhores práticas na empresa;
- É flexível e pode se adaptar às mudanças da empresa, desde que corretamente personalizado;
- Permite a colheita de dados de forma automatizada;
- É possível ter acesso aos dados e informações em tempo real;
- Estrutura modular, o que permite uma implantação gradual em algumas áreas de cada vez;
- Padronização dos sistemas de informação;
- Os controles operacionais são melhorados;
- Transparência e confiabilidade da informação;
- As diversas áreas da empresa se tornam mais integradas, quebrando nichos;
- Informações integradas, gerando “uma única fonte da verdade”;
- Facilita enormemente a tomada de decisões estratégicas e táticas.

Quanto às desvantagens, a que mais se destaca é, sem dúvida, os custos de toda a implementação deste tipo de sistemas (planeamento, testes, configuração, personalização, entre outros). A personalização pode ser demorada e dispendiosa e, caso seja mal executada, pode ter um impacto bastante negativo na atividade da empresa.

### **3.3 Systems Applications and Products (SAP)**

A empresa SAP é uma das maiores empresas de *software* de negócios do mundo. Tem um conjunto alargado de produtos de *software* de gestão, com os quais oferece soluções dedicadas a cada cliente.

O seu principal produto é o *software* de ERP. SAP ERP é um *software* de planeamento de recursos empresariais desenvolvido pela empresa alemã SAP SE. Este *software* incorpora as principais funções comerciais de uma organização. O sistema procura contemplar a empresa como um todo, dividindo-se em módulos, cada um correspondente a uma área específica, como por exemplo, o módulo SD (*Sales and Distribution*) que contempla a área de Vendas e Distribuição, fazendo a integração das informações para um determinado processo. [15] [16] [17] A figura 2 ilustra a lógica de um sistema ERP.

Cada programa é executado através de uma transação independente. Estes programas são desenvolvidos em ABAP, uma linguagem de programação, na qual a SAP detém os direitos.



Figura 2 - Lógica de um sistema ERP [18].

### 3.3.1 Módulos SAP

Cada módulo é responsável por mais de mil processos de negócios, baseado em práticas do dia-a-dia de cada empresa. O sistema é configurado para atender às necessidades de cada processo, onde mais de 8 mil tabelas administram em tempo real as informações que são usadas pela empresa. [19]

Os principais módulos presentes neste sistema são os descritos na tabela 1.

Tabela 1 - Módulos de SAP ERP [19]

<b>Módulos</b>	<b>Descrição</b>
<b>MM</b>	<i>Material Management</i>
<b>WM</b>	<i>Warehouse Management</i>
<b>SD</b>	<i>Sales and Distribution</i>
<b>FI</b>	<i>Financial Accounting</i>
<b>PP</b>	<i>Production Planning and Control</i>
<b>HCM</b>	<i>Human Capital Management</i>
<b>PS</b>	<i>Project System</i>
<b>CO</b>	<i>Controlling</i>
<b>QM</b>	<i>Quality Management</i>
<b>PM</b>	<i>Plant Maintenance</i>
<b>IS</b>	<i>Industry Solutions</i>
<b>BW</b>	<i>Business Warehousing</i>
<b>RE</b>	<i>Real Estate</i>

### 3.3.2 Fluxo de Trabalho entre Equipas

A empresa SAP contém três áreas de formação:

- Administradores de sistema – Responsáveis pela gestão da plataforma SAP, controlo de desempenho e passagem de programas entre os ambientes de desenvolvimento, testes e qualidade e produção;
- Consultores técnicos – Responsáveis por estender e criar funcionalidades na plataforma SAP (possuem conhecimentos e formação em ABAP ou em Java);
- Consultores funcionais – Responsáveis por implementar regras de negócio e fluxos de informação.

Quando o *software* ERP nativo da SAP não possui funcionalidades que permitam satisfazer as necessidades solicitadas pelos seus utilizadores, é elaborado um projeto de desenvolvimento para uma equipa de SAP.

Os consultores funcionais elaboram especificações para que os consultores técnicos as possam desenvolver e implementar. Estas especificações podem decorrer de



pedidos dos utilizadores e/ou alterações processuais devido a modificações legais ou novas necessidades. Este fluxo está demonstrado na figura 3.

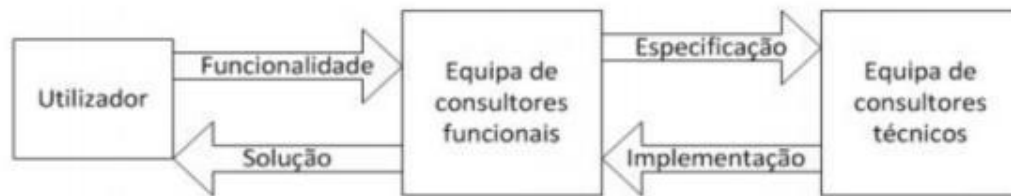


Figura 3 - Fluxo de trabalho entre equipas.

As especificações técnicas, além de explorarem o problema, podem apresentar trabalho já desenvolvido. As implementações feitas pelos consultores técnicos, podem ser módulos de função, procedimentos e outros objetos, cujo nome começa obrigatoriamente por “Z” ou “Y” para indicar que não é um programa *standard* da SAP. Num *upgrade* de sistemas por exemplo, a SAP consegue identificar os programas desenvolvidos por terceiros, evitando que os mesmos sejam alterados ou apagados da mesma forma que os programas *standard*.

Após ser implementada a especificação técnica, o consultor técnico responsável regista o seu trabalho no documento que recebeu do consultor funcional.

A plataforma SAP é modular e o desenvolvimento é feito através da implementação de processos. Alguns conceitos fundamentais são:

- Processo – É uma sequência lógica de procedimentos e fluxos funcionais incluindo a recolha de informação, o seu processamento, e resultados deste;
- Transação – É uma instanciação de um processo que pode resultar em sucesso ou erro. As transações são únicas e apenas uma transação instancia um processo também chamado como programa;
- Telas – São os ecrãs para o utilizador, podendo ser desenvolvidas ou criadas pelo próprio sistema;
- Programas – são os módulos onde se implementam funções de processamento.

### 3.4 Advanced Business Application Programming (ABAP)

ABAP (*Advanced Business Application Programming*), é uma linguagem de programação de alto nível usada no desenvolvimento e personalização em *software* SAP. [24]

SAP ABAP é a linguagem mais usada no desenvolvimento de aplicações empresariais para grandes negócios e instituições financeiras em SAP R/3. O SAP R/3 é um *software* de gestão de recurso nas empresas (ERP).

Como se ilustra na figura 3, os sistemas SAP utilizam uma arquitetura de nível 3 Cliente/Servidor. É descrita da seguinte forma.

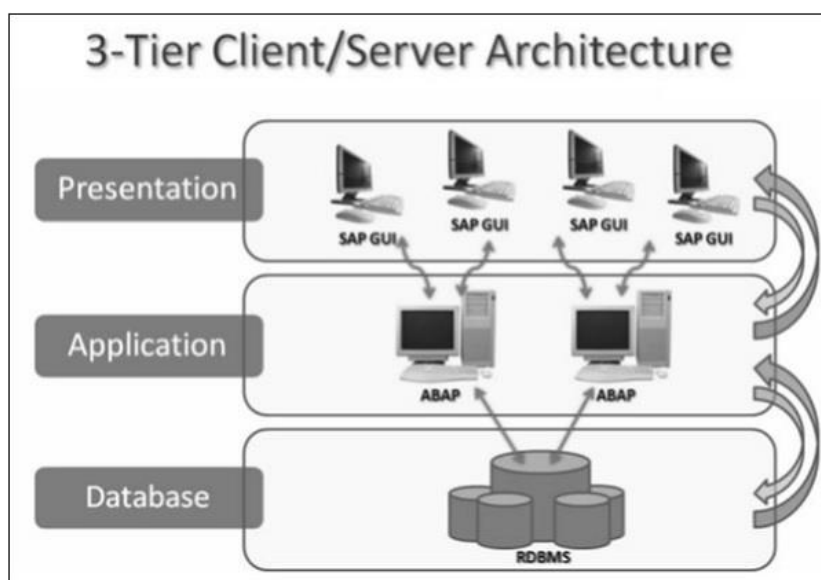


Figura 4 - Arquitetura do sistema ABAP [24].

A camada de apresentação (ou Acesso) consiste em qualquer tipo de entradas que possam ser usadas para controlar um sistema SAP, isto pode ser uma página *web*, um dispositivo móvel, entre outros. Todo o processo central ocorre na camada de aplicação. A camada de aplicação não é só um sistema em si, mas pode conter várias instâncias do sistema de processamento. O servidor comunica com a camada de Base de Dados, que normalmente é mantida num servidor à parte. Os programas ABAP são executados ao nível da camada de Aplicação.

O ABAP é uma linguagem que obriga a conhecimentos avançados de programação e é utilizada pelos programadores da SAP, que desenvolvem o *software*, e por consultores ABAP, que adaptam o *software* às necessidades dos clientes. A sua sintaxe é similar à do COBOL (linguagem de programação orientada para o processamento de base de dados comerciais).

### 3.5 SAP Fiori

O SAP Fiori é uma nova abordagem da SAP que permite o desenvolvimento de aplicações para *browsers* ou dispositivos móveis. O SAP Fiori oferece mais de 300 aplicações *standard* que podem ser usadas em diversas Áreas de negócio (Fig. 5), como recursos humanos, manutenção de plantas, finanças, etc. [25]

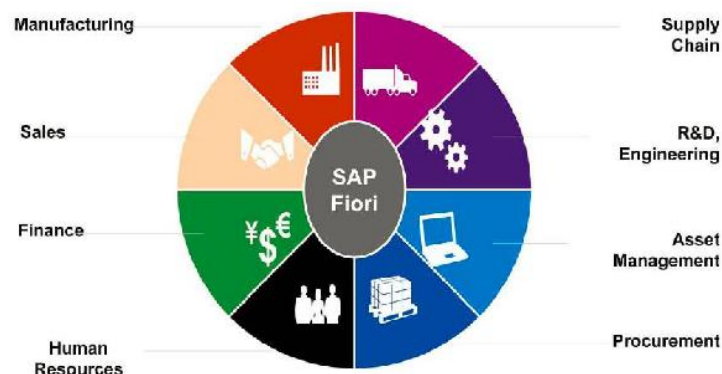


Figura 5 - SAP Fiori [25].

As aplicações de SAP Fiori podem ser utilizadas em qualquer computador e permitem a continuidade do mesmo processo em qualquer dispositivo móvel, como se ilustra na figura 6.



Figura 6 – Launchpad do Fiori [25].

O uso de aplicações SAP Fiori tem inúmeras vantagens como [26]:

- Aumentar a produtividade nas empresas, ao simplificar os seus processos;
- Elevada mobilidade;
- Pode ser utilizado em diversos dispositivos, de diferentes formas e tamanhos;
- Aumenta a aceitação por parte dos colaboradores;
- Reduz tempo de formação e adaptação dos utilizadores às aplicações.

### 3.6 Neptune

O Neptune é um *addon* para sistemas SAP, desenvolvido pela empresa Neptune Software. Este *addon* permite o desenvolvimento de aplicações para computador e qualquer dispositivo móvel com integração direta nos sistemas SAP. É líder de desenvolvimento rápido de aplicações de baixo código que ajuda a modernizar e otimizar processos de negócios e interfaces de utilizador. Esta ferramenta, para o desenvolvimento base de aplicações, usa as linguagens de programação Javascript e CSS para o *front-end* e ABAP para o *back-end*. Outras tecnologias que se podem também usar são Node.js, Adobe PhoneGap, SQLite, IndexedDB e JSONH [27] [28] [29] [31].

As suas aplicações podem ser compiladas para iOS, Android e Windows 10 com acesso total aos dispositivos e às suas capacidades.

O Neptune é a única plataforma de desenvolvimento de aplicações com certificação SAP e permite que seja possível otimizar e modernizar os processos SAP e as interfaces dos utilizadores, tendo todos os benefícios do SAP Fiori e ABAP.

Através do Neptune, também é possível fazer uma cobertura de todo o ciclo de vida de uma aplicação desenvolvida nesta plataforma. O ciclo de vida da aplicação é definido em três fases: Desenvolvimento, Administração e Acesso (Fig. 7).

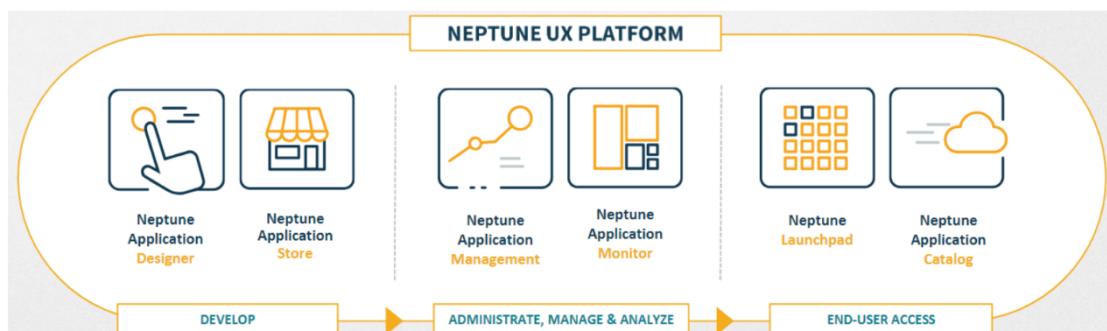


Figura 7 - Ciclo de vida de uma aplicação em Neptune [28].

**Desenvolvimento** - O programador pode desenvolver as aplicações de raiz, através de *drag and drop* de objetos para a aplicação (Fig. 8). O utilizador pode fazer a programação em Javascript nesses objetos conforme necessário. O programador poderá também ir à loja Neptune Application Store onde pode obter aplicações demo e desenvolver por cima das mesmas, poupando assim muito tempo. Nem sempre é a melhor alternativa a seguir, pois é uma opção que depende bastante do tipo de aplicação e do tempo estipulado para o seu desenvolvimento. As aplicações demo apenas deveriam de servir como demonstração pois contém alguns erros e não fazem o trabalho que é necessário desenvolver. O programador também pode aceder a uma biblioteca de documentação sobre cada objeto específico e disponível para ser usado. Esta documentação fornece-nos dados sobre cada objeto, disponibiliza exemplos demonstrativos sobre o objeto e um conjunto de propriedades do objeto que indica se o objeto está em uso e tudo o que pode ser feito com ele.





Figura 10 - Launchpad do Neptune [30].

Esta solução sofreu recentemente uma atualização (*rebranding*), e neste momento, a Neptune Software disponibiliza duas soluções: Planet 8 (antigo Neptune), que é usado em SAP e o Planet 9 que tem como base o uso de APIs. Ambas as soluções são muito semelhantes a nível da arquitetura, permitindo que qualquer programador de Planet 8 possa desenvolver aplicações em Planet 9 sem grandes adaptações e dificuldades.

O fluxo de dados de aplicações Neptune (Fig. 11) utiliza um padrão de desenho de desenvolvimento MVC. As visões de Neptune são desenhadas em HTML e CSS. Também podem ter funcionalidades de controladores desenvolvidas em JavaScript. O controlador é uma classe de ABAP dividida por métodos. Os métodos realizam as várias funcionalidades da aplicação, atualizando e pesquisando dados na base de dados para serem disponibilizadas na visão.

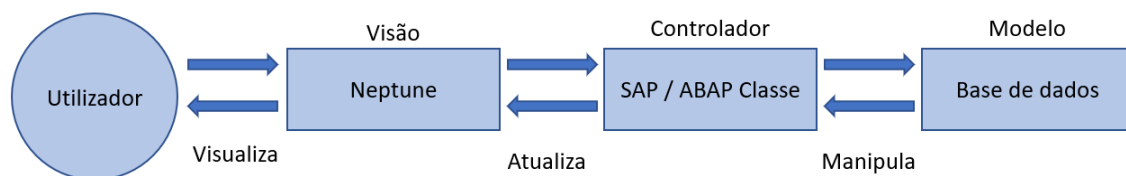


Figura 11 - Fluxo de dados simples numa aplicação Neptune.

## 3.7 SAP Leonardo

SAP Leonardo é um sistema digital inovador que ajuda os seus clientes a aceder a componentes e produtos SAP através de tecnologias emergentes como *Machine Learning*, *Big Data*, *Internet of Things* e *Blockchain*. Está também muitas vezes associado à plataforma Cloud da SAP mas não está limitado a apenas isso, porque abrange todas as tecnologias modernas e inclui capacidades de previsão para S/4HANA seja localmente ou na nuvem. [32] [33] [34]

SAP Leonardo, também está focado no *design thinking*, nas tecnologias de encapsulamento e na definição de metodologias para enfrentar desafios de negócios. O *design thinking* é uma metodologia que se concentra em ideias e soluções criativas para resolver *use-cases*. A premissa é que os clientes não precisam apenas de boa tecnologia, eles precisam de uma abordagem para descobrir o valor de como e o que a tecnologia pode fazer por eles.

Um dos principais benefícios do SAP Leonardo é que ele permite integrar vários produtos SAP com tecnologias emergentes. Por exemplo, no SAP Cloud Platform, pode-se criar aplicações Fiori que usam serviços IoT da SAP. Quando integrado nos processos de negócios o utilizador final não tem a perceção da existência dos algoritmos de *Machine Learning*, mas sem o saber, fará uso desta potente tecnologia quando experienciar os seus benefícios na tomada de decisões ou recomendações que são produzidas automaticamente.

### 3.7.1 Soluções SAP Leonardo

Neste momento, o SAP Leonardo é uma plataforma que disponibiliza um conjunto de tecnologias (Fig. 12) que permitem auxiliar as empresas tanto na gestão como no apoio à tomada de decisão. [33]



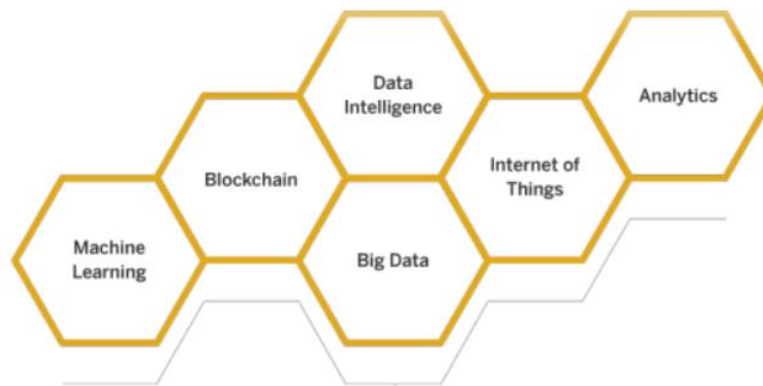


Figura 12 - Tecnologias que constituem o SAP Leonardo [33].

As tecnologias de *Machine Learning* (ML) permitem construir uma empresa inteligente através de algoritmos de inteligência artificial (IA) que une o conhecimento humano e as percepções do computador.

*Internet of Things* (IoT) permite ligar digitalmente um negócio entre pessoas e processos. Com o IoT pode-se recolher e analisar dados a partir de sensores e dispositivos inteligentes que podem ser usados para transformar modelos de negócios, processos e operações. [34]

*Analytics* permite obter informações rápidas e precisas em toda a organização com ferramentas avançadas para visualização de dados, planeamento, análise preditiva e *Big Data*.

*Blockchain* permite simplificar e otimizar processos multipartidários complexos e reduzir a dependência de intermediários com a tecnologia *blockchain* e *ledger* distribuído.

*Data Intelligence* permite criar informação a partir dos dados recolhidos, criar novos fluxos de receita e tornar o negócio realmente orientado a dados com serviços de *Data Intelligence*.

*Big Data* permite usar uma base de dados em memória e ferramentas mais recentes em gestão de *Big Data* para unificar todos os dados em um cenário confiável para alimentar aplicações e serviços de *Analytics*.

### 3.8 SAP HANA

O SAP HANA (*High-Performance Analytic Appliance*) é uma tecnologia de base de dados em memória que permite o processamento de grandes quantidades de dados em tempo real e num curto período de tempo. O SAP HANA processa dados armazenados na RAM em vez de os ler a partir de um disco, permitindo fornecer resultados instantâneos de transações de clientes e análises de dados. [35] [38]

O SAP HANA tem uma componente de programação que permite às empresas criar e executar programas personalizados sobre o HANA, bem como um conjunto de bibliotecas de análise preditiva, espacial e de análise de texto (Fig. 13). O HANA pode ser executado em paralelo com um programas SAP ERP, os analistas podem aceder a dados para processamento analítico em tempo real e não precisam de esperar pela execução de um relatório diário ou semanal. [36] [38]

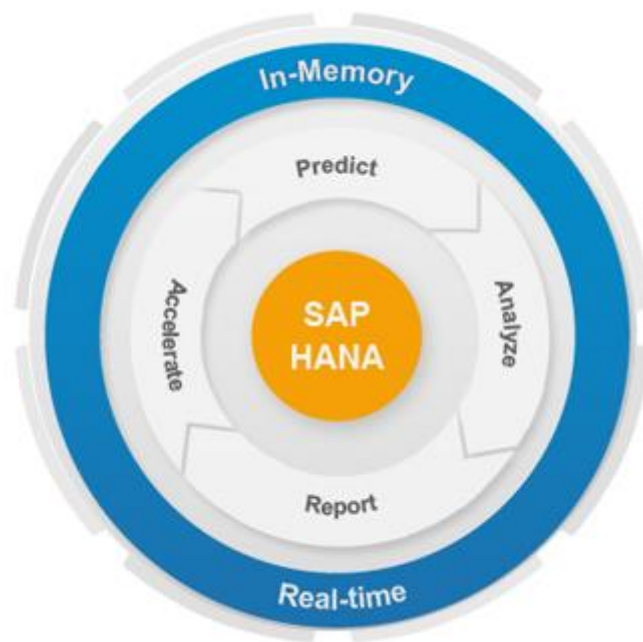


Figura 13 - Funcionamento do SAP HANA [37].

## Capítulo 4 | Metodologias de projeto

Neste capítulo são descritas as metodologias de trabalho empregues no desenvolvimento de projetos e do *software* utilizado no desenvolvimento de soluções SAP ABAP e Neptune. É feita uma introdução às metodologias ágeis e tradicionais, metodologia SCRUM, metodologia *Waterfall* e no final é feita uma avaliação comparativa destas metodologias [39].

### 4.1 Breve introdução

A utilização de metodologias de projeto associadas à adoção das melhores práticas de gestão é essencial para que uma empresa consiga planejar e concretizar os seus objetivos e estratégias, assegurando uma forte vantagem competitiva [40] [41].

Existem vários tipos de metodologias, e a sua utilização depende de projeto para projeto. Usar uma metodologia permite a execução adequada de todas as atividades envolvidas no processo e garante a eficiência do projeto. É importante contar com ferramentas que se alinhem com as premissas da empresa, melhorando os seus resultados e unificando os principais setores [40] [41].

A metodologia a ser utilizada em cada projeto, depende tanto do gestor do projeto como do cliente, são eles que definem a metodologia a ser utilizada, existindo assim alguns aspetos levados em consideração como os objetivos do projeto, prazos para a finalização do mesmo, riscos envolvidos, custos, entre outros [40] [41].

No decorrer do estágio, nas tarefas e no projeto realizado foram utilizadas tanto as metodologias ágeis como as tradicionais, sendo que as metodologias mais utilizadas no desenvolvimento de soluções SAP são o SCRUM e a *Waterfall*. As descrições destas metodologias são descritas nos sub-capítulos seguintes [40] [41].

## 4.1 Metodologias Ágeis e Tradicionais

A metodologia ágil faz uma abordagem diferente ao modelo de gestão tradicional de projetos que definia o desenvolvimento de produtos dividido por etapas bem definidas. Já os Métodos Ágeis possuem iterações mais curtas e incrementais, que são continuamente reavaliadas e se necessário adaptadas com o objetivo de melhorar os resultados da gestão das equipes e processos [42]. A origem dos Métodos Ágeis está normalmente associada com a abordagem da Toyota ao sistema “*Lean Manufacturing*” e às novas abordagens para a gestão de projetos de *software* conhecido como Manifesto Ágil (2001). Ilustrado na figura 14, estão diversas metodologias ágeis [42].

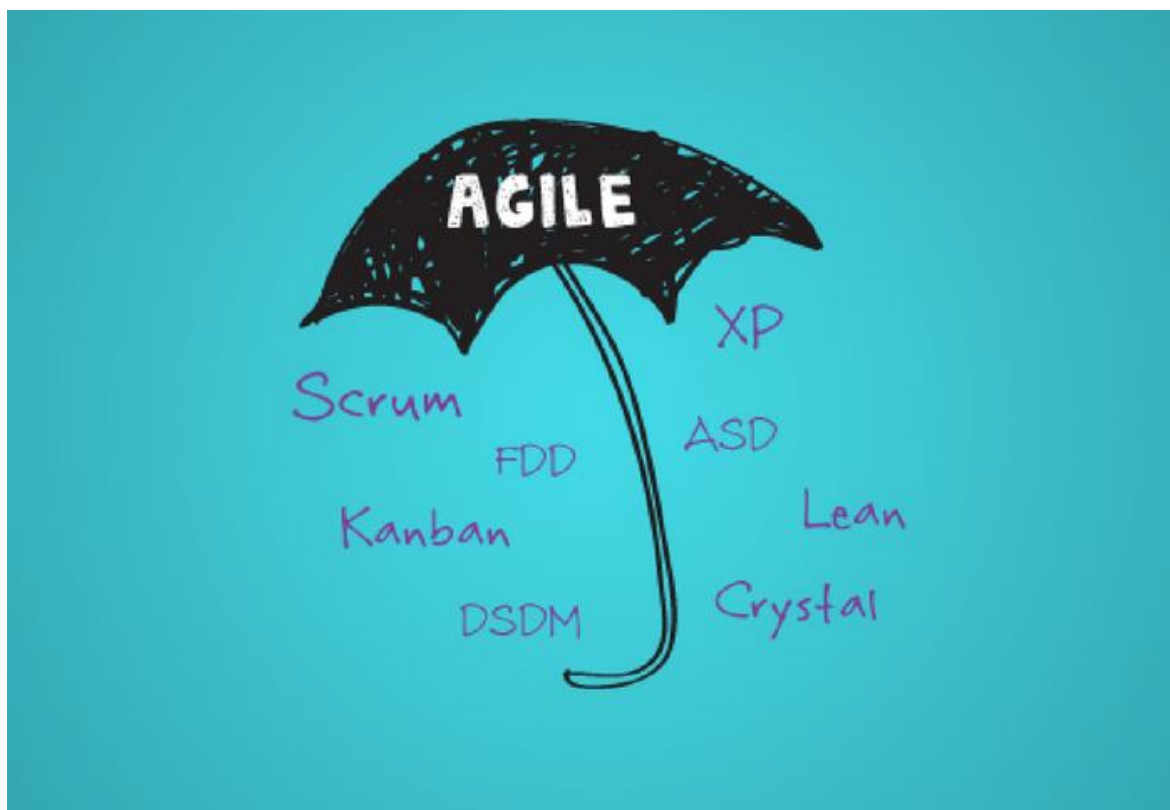


Figura 14 - Breve visão das metodologias Ágeis [42]

A metodologia ágil tem uma abordagem mais dinâmica. O desenvolvimento dos processos são alinhados a um programa colaborativo, concentrado em velocidade e na flexibilidade. A metodologia ágil utiliza ferramentas, com grande capacidade de atuação. Os

profissionais, tal como as ferramentas, trabalham de modo interativo, envolvendo grande participação da equipa nas tarefas [40] [41].

Utilizando uma metodologia ágil, podem ser entregues aos clientes funcionalidades mínimas de um projeto antes de estar totalmente concluído, podendo ser dimensionado em etapas ou entregas parciais. Este procedimento facilita a resolução inteligente de programas, evitando a existência de erros e falhas decorrentes do processo [40] [41].

A metodologia tradicional trabalha de forma mais hierarquizada. A tradicional opera com modelos de procura conhecidos como “cascata”, nos quais as etapas são mais rígidas e programadas. O processo é sequencial e focado no resultado final, não havendo abertura para implementações em conjunto com o cliente durante o desenvolvimento [40] [41].

Considerada uma metodologia de trabalho ultrapassada, pode ser importante para estruturar planeamentos em empresas que operam com equipas grandes e projetos e tarefas muito específicas [40] [41].

## 4.2 Metodologia SCRUM

SCRUM é uma metodologia utilizada na gestão de projetos e desenvolvimento ágil de *software*. O Scrum é uma metodologia ágil que pode ser utilizada em conjunto com a gestão de projetos. Esta metodologia ágil engloba um conjunto de práticas focadas na rapidez e na eficiência das ações de uma equipa, no uso de *softwares*, na participação do cliente e na capacidade de reagir de forma madura às principais mudanças. Segundo o “*State of Agile*” [43] é o método ágil mais utilizado no mundo com quase 80% dos projetos [44] [46].

Esta metodologia funciona dividindo o projeto em pequenos passos chamados *sprints*. O *sprint* representa um intervalo de tempo no qual um conjunto de atividades deve de ser concluído [43].

No fim de cada *sprint*, a equipa deve realizar uma breve reunião com o objetivo de se fazer uma retrospectiva, na qual apresentam os resultados e se discute o que poderia ter acontecido de maneira diferente. Deve-se apontar os fatores de sucesso e as principais interferências encontradas, replaneando as atividades para o próximo *sprint* e deste modo

procurar a melhoria contínua. Ilustrado na figura 15, é demonstrado o funcionamento da metodologia SCRUM [43].

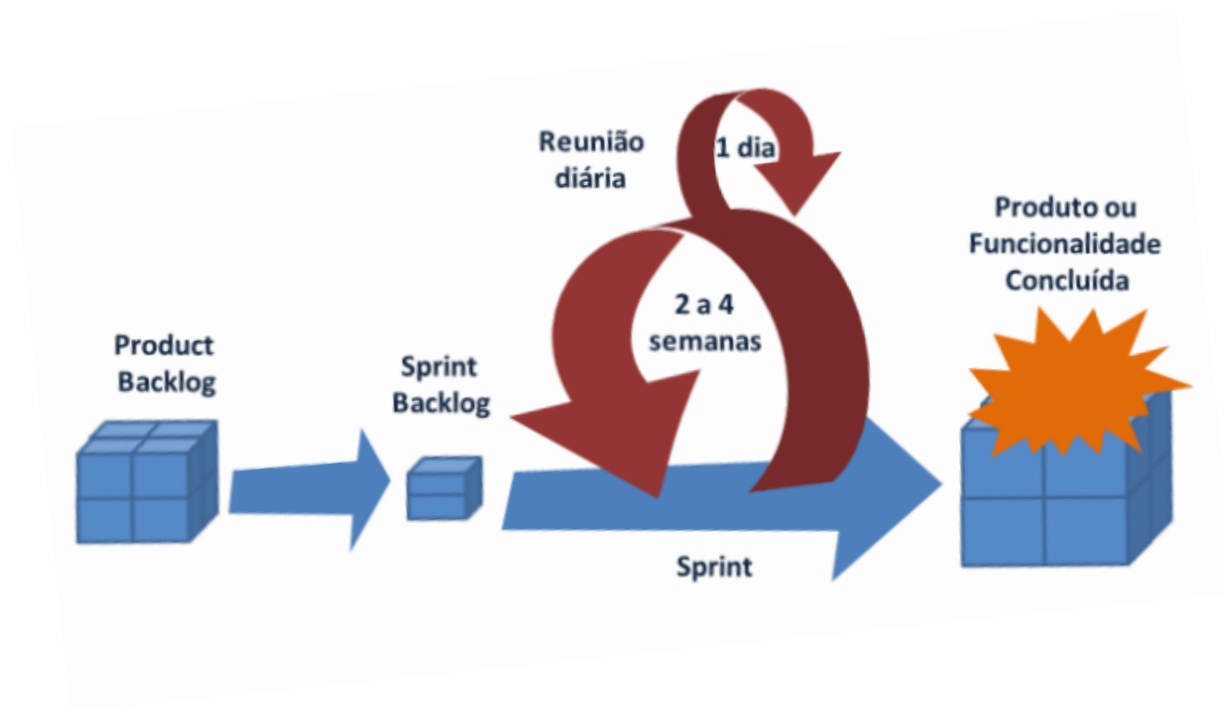


Figura 15 - Fluxo de trabalho utilizando a metodologia SCRUM [45]

### 4.3 Metodologia *Waterfall*

*Waterfall* é uma metodologia tradicional de gestão de projetos. O seu principal objetivo é a execução sequencial de etapas, a partir de uma ordem lógica, procurando atingir um determinado resultado [47] [49].

Esta metodologia é simples de se entender, porque uma tarefa só pode ser começada quando sua antecessora for concluída. A sua desvantagem é que qualquer alteração, por menor que seja, no objetivo do projeto, pode interromper o processo e causar grandes complicações.

Apesar de ser considerada uma metodologia tradicional e amplamente empregada em setores de fabricas ou construção, não é considerada flexível.

A figura 16 demonstra o funcionamento da metodologia *Waterfall*.

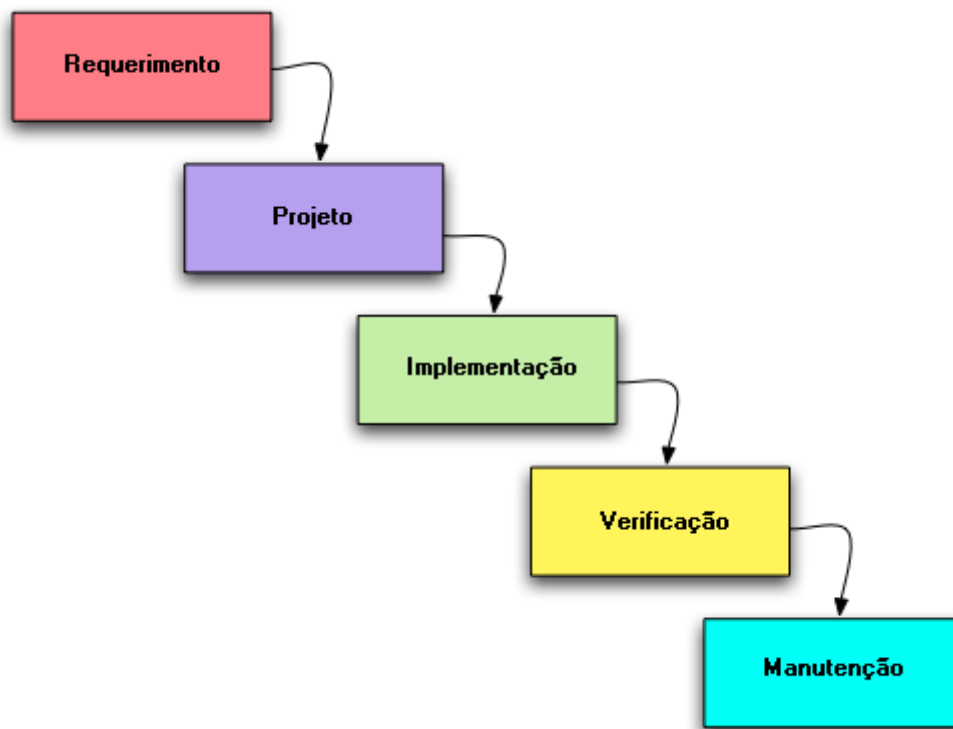


Figura 16 - Fluxo de trabalho utilizando a metodologia *Waterfall* [47]

## 4.3 Comparação de metodologias

Na tabela 2, é apresentado um resumo da avaliação das metodologias estudadas.

Tabela 2 - Diferenças entre metodologias tradicionais e as metodologias ágeis [48] [50].

Metodologias Tradicionais	Metodologias Ágeis
Planeamento rígido.	Maior liberdade no planeamento das ações.
Resistência a mudanças.	Flexibilidade e uma postura positiva diante da necessidade de mudanças (mesmo em fases finais do projeto).
Decisões tomadas numa abordagem <i>top-down</i> .	Liberdade para tomar decisões em equipa.
Forte centralização em torno da figura do Gestor de Projeto.	Responsabilidade compartilhada entre todos os membros da equipa e espírito de colaboração.
Uma liderança que monopoliza toda a comunicação já que está preocupada com o comando e o controlo detalhado das ações.	Comunicação fluída e livre entre os membros da equipa.
Líderes indicam “O que fazer”, “Como”, em vez de dizer “Porquê”	Equipas auto-organizáveis; a divisão do trabalho é resultado do entendimento do projeto e de um consenso entre a equipa.
Problemas geralmente escalados até à gestão.	Atuação conjunta da equipa para a resolução de problemas.
O planeamento costuma prever um trabalho extenso, com a entrega do produto somente nos estágios finais do cronograma (o que leva a muitos conflitos com o cliente).	Entregas de partes do projeto de forma contínua e incremental (iterações), a fim de se obter um rápido <i>feedback</i> do cliente acerca do andamento do projeto.
Uma longa fase de análise; em muitos casos, parte da equipa é deixada de lado nestes estágios iniciais (já que considera que tais membros ingressarão apenas na fase de execução).	Reuniões diárias entre a equipa; o intuito está em discutir o que será feito naquele momento, revendo o planeamento a médio e curto prazo, além de prováveis impedimentos.
Um grande foco na criação de documentos.	Embora existam documentos e se estimule a criação dos mesmos, existe um pragmatismo maior.
Um maior foco nos processos do que no produto esperado.	Menos formalidade e maior ênfase em se chegar ao produto esperado.
Maior envolvimento do cliente em estágios iniciais, com um certo relaxamento de postura assim que o projeto tenha início.	Participação ativa do cliente, inclusive enquanto o projeto está a ser implementado.
Foco na “antecipação” (tarefa difícil num ambiente sempre sujeito a mudanças repentinas).	Ênfase na “adaptação” (“jogo de cintura”).



## Capítulo 5 | Apresentação e descrição das atividades realizadas

### 5.1 Atividades Realizadas

Neste capítulo é apresentado um projeto desenvolvido durante o estágio, e a implementação de notas que visam a correção e melhoria de um sistema, onde é explicado em que consiste, os seus objetivos, e os procedimentos necessários para a sua realização.

O projeto baseado em programação Neptune, começou por ser usado em aplicações genéricas para demonstração a clientes, contudo, acabou por se tornar num produto comercial que foi vendido a um cliente.

A implementação de notas, consiste na instalação de um conjunto de correções a fim de melhorar um sistema.

### 5.2 Caso 1 | Projeto

#### 5.2.1 Projeto

A pedido da Softinsa, foi solicitado a criação de aplicações sobre o modulo PM (*Plant Maintenance*) para uso genérico na sua máquina *sandbox*. Estas aplicações tinham o objetivo inicial de fazer demonstrações a clientes e futuramente, torná-lo num produto comercial, ajustando-o aos interesses do cliente. As aplicações deveriam também de estar adaptadas a funcionar em modo *mobile*, *online*, *offline* e deveriam ter descrições em português, mas com traduções para inglês e espanhol.

As aplicações sobre o modulo PM servem para fazer a manutenção de problemas em armazéns. Para auxílio à manutenção, existem dois tipos de documentos, as notificações e as ordens de trabalho.

As notificações são documentos que informam ao operário o que há para resolver no armazém. Cada notificação pode ter uma ordem de trabalho.

As ordens de trabalho são os documentos que vão indicar as tarefas que o operário tem para resolver. São divididas principalmente por dados mestre, operações e componentes.

Os dados mestre indicam qual o local de instalação, o equipamento a ser reparado e a prioridade da reparação. As operações a serem realizadas informam que tipo de atividade será realizado, e quanto tempo e operários são necessários. As componentes é o material necessário para fazer o trabalho.

As ordens de trabalho podem ter *status* do objeto que indicam o estado do documento e o *status* de usuário que indica em que fase do processo está o operário. Também podem ser liberadas, mudando o *status* do documento para liberado. As ordens liberadas também têm confirmações onde se valida o tempo real que as operações levaram a ser realizadas e que componentes foram gastas ou usadas. Ordens liberadas e confirmadas podem ter o seu *status* mudado para terminado, concluindo o seu processo de uma ordem de trabalho.

A metodologia utilizada para realizar este projeto foi baseada em SCRUM, pois o projeto seria dividido em várias aplicações, cada uma com uma tarefa específica para ter sempre algo novo e a funcionar a cada iteração do projeto.

Neste projeto, o programador iria trabalhar em conjunto com um colaborador funcional. O dever do funcional, será especificar, planejar e ditar o que as aplicações deveriam de fazer.

### **5.2.2 Análise de Requisitos para aplicações de demonstração**

Devido à já existência de aplicações de demonstração da Neptune Software, foi pedido a realização de uma copia das mesmas para as adaptar, conforme especificado, e para cumprir com os requisitos estabelecidos pelo funcional.

Foi pedido a copia para adaptação de 5 aplicações (Fig. 17) para a máquina *sandbox* da Softinsa:

- Listagem e detalhes de equipamentos.
- Listagem e detalhes de locais de instalação.
- Listagem e detalhes de ordens de trabalho.
- Listagem, CRUD e funcionalidades de PM para notificações.
- Listagem, CRUD e funcionalidades de PM para ordens de trabalho.

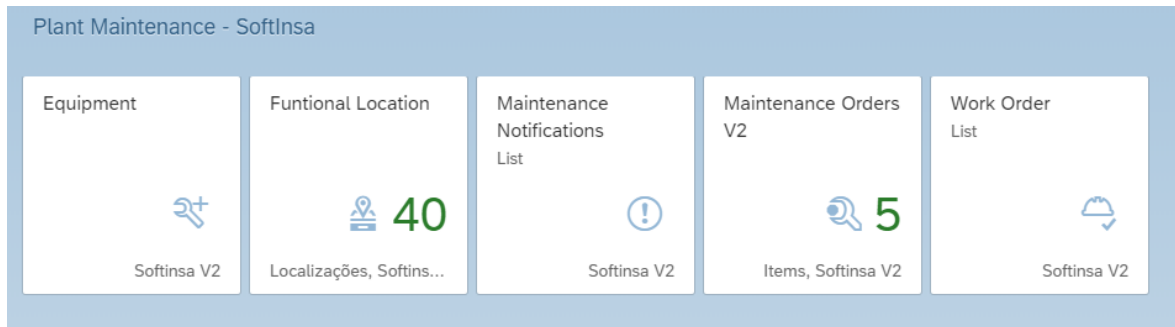


Figura 17 - Tiles das aplicações genéricas no *launchpad* na máquina *sandbox* da Softinsa.

De notar, que algumas aplicações, são dependentes de outras aplicações. Por exemplo, a aplicação de listagem, CRUD e funcionalidades de PM para ordens de trabalho é na realidade um conjunto de varias aplicações (Fig. 18). Começando por uma aplicação de listagem de ordens, esta faz ligação a uma aplicação de CRUD para um documento de ordem, que por sua vez, faz ligação a uma aplicação de criação de operações, criação de componentes, e criação de anexos. A aplicação de listagem e detalhes de equipamentos também faz ligação com a aplicação de CRUD para um documento de ordem, mas irá ter outro comportamento devido à aplicação origem de que é chamada. Se for chamada do *launchpad* de aplicações, irá ter um comportamento de listagem e detalhes. Se for chamada da aplicação de CRUD das ordens, irá apenas listar e não detalhar.

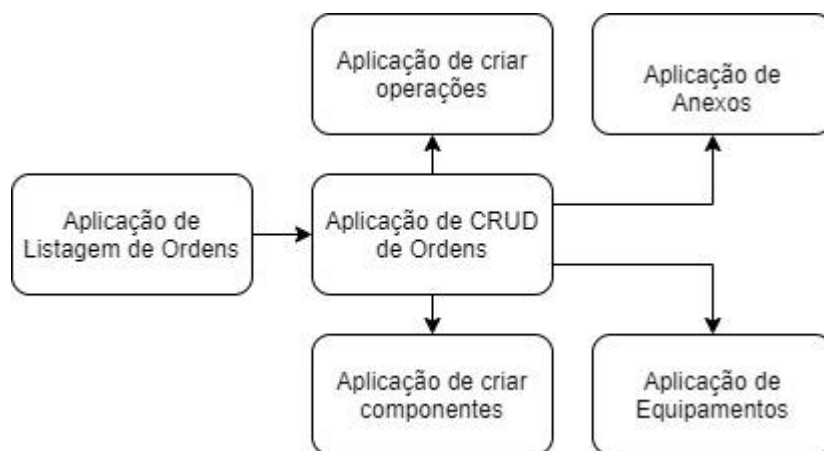


Figura 18 - Dependencia da aplicação de listagem de ordens com CRUD.

Para a adaptação das aplicações ao modo *mobile*, algumas partes dos ecrãs e tabelas foram ajustados para que sejam escondidos ou não contenham muita informação para se conseguir compreender o que mostram.

O mecanismo que foi implementado para as aplicações funcionarem em modo *offline*, foi a utilização de memória *cache* utilizando a tecnologia IndexedDB. Ao abrir qualquer aplicação pela primeira vez, seriam descarregados todos os dados necessário para o dispositivo para que pudessem ser acedidos em modo *offline*.

#### **5.2.2.1 Aplicação de equipamentos**

A aplicação de listagem e detalhes de equipamentos apresentava as seguintes especificações:

1. O ecrã principal, deverá estar dividido em três partes (Fig. 19).
  - a. À esquerda, em cima, campos de seleção para ajudar na pesquisa de equipamentos;
  - b. À esquerda, por baixo do filtro, uma listagem de equipamentos;
  - c. Do lado direito, depois de escolher um equipamento, os seus detalhes.
2. Esta aplicação terá apenas a funcionalidade do ecrã à esquerda, quando for chamada a partir de uma aplicação de CRUD de um documento de notificação ou ordem de trabalho. Irá servir como ajuda de pesquisa e preenchimento automático do campo de equipamento depois do utilizador escolher um (Fig. 20).

Seleção e Listagem

Seleção

Número de Equipamento:

Descrição:

Pesquisa

Listagem

10000000

Equipment\_01\_TST

10000001

Maquina de enchimento

10000002

Equipamento PR-A

10000003

Equipamento PR-B

10000004

Equipamento PR-C

10000005

Equipamento Q-A

Excluir

Lista de resultados

Equipment\_01\_TST

Status: AEQS

Válido de: 14.09.2018

Válido até: 31.12.9999

Gerar

Localização

Estrutura

Classes

Pontos de m...

Tipo de objeto

003

Climatizacao

Peso

500 g

Inventário

123098

Em serviço desde

20180912

Fabricante

Fábrica

Denominação

AR

Número de série

123456789

Excluir

Fechar

Figura 19 - Aplicação genérica de listagem e detalhes de equipamentos.

Seleção e Listagem

Seleção

Equipamento:

Descrição:

Pesquisa

Listagem

10000000

Equipment\_01\_TST

10000001

Maquina de enchimento

10000002

Equipamento PR-A

10000003

Equipamento PR-B

Excluir

Figura 20 - Aplicação genérica de listagem de equipamentos sem detalhes.

Mais tarde foi pedido um novo desenvolvimento para a ajuda de pesquisa de equipamentos. Este novo desenvolvimento foi a criação de um botão para aceder à câmara do dispositivo móvel, utilizando uma funcionalidade de leitura de códigos de barras como apresentado nas figuras 21 e 22. Este botão apenas estaria visível se o utilizador utilizasse um dispositivo móvel.

The screenshot shows a mobile application window titled "Seleção e Listagem" with a close button (X) in the top right corner. The interface is divided into two main sections: "Seleção" and "Listagem".

**Seleção Section:**

- Contains two input fields: "Equipamento:" and "Descrição:".
- Next to the "Equipamento:" field is a button with a barcode icon, representing the mobile device camera for barcode scanning.
- Below the input fields is a large blue button labeled "Pesquisa".

**Listagem Section:**

- Contains a large rectangular area with the text "Sem dados", indicating no data is currently displayed.

**Footer:**

- A dark blue bar at the bottom contains a button labeled "Excluir".

Figura 21 - Aplicação genérica de listagem de equipamentos sem detalhes modo *mobile* (tablet) com botão de leitura de código de barras.



Figura 22 - Câmara em modo de leitura de código de barras.

A aplicação deve seguir o diagrama de casos de uso como na figura 23.

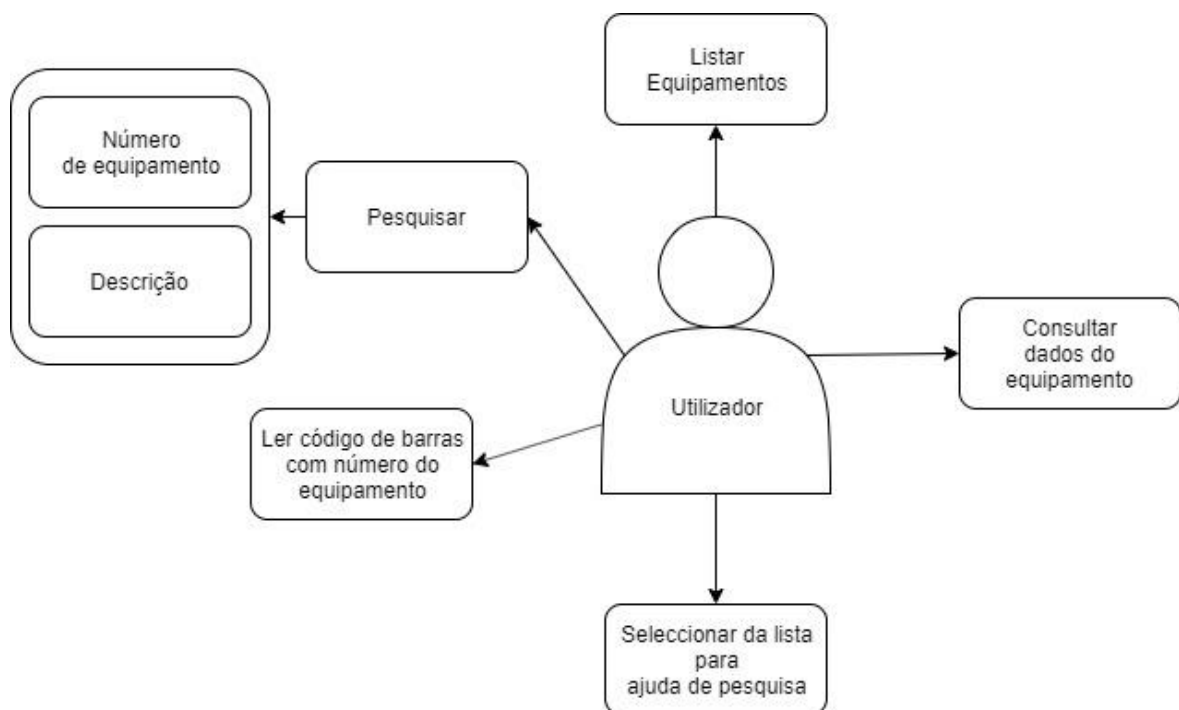


Figura 23 - Diagrama de casos de uso para aplicação de listagem de equipamentos.

### 5.2.2.2 Aplicação de locais de instalação

A aplicação de listagem e detalhes de locais de instalação apresentava as seguintes especificações:

1. O ecrã principal, deverá estar dividido em três partes (Fig. 24).
  - a. À esquerda, em cima, campos de seleção para ajudar na pesquisa de locais de instalação;
  - b. À esquerda, por baixo do filtro, uma listagem de locais de instalação;
  - c. Do lado direito, depois de escolher um locais de instalação, os seus detalhes.

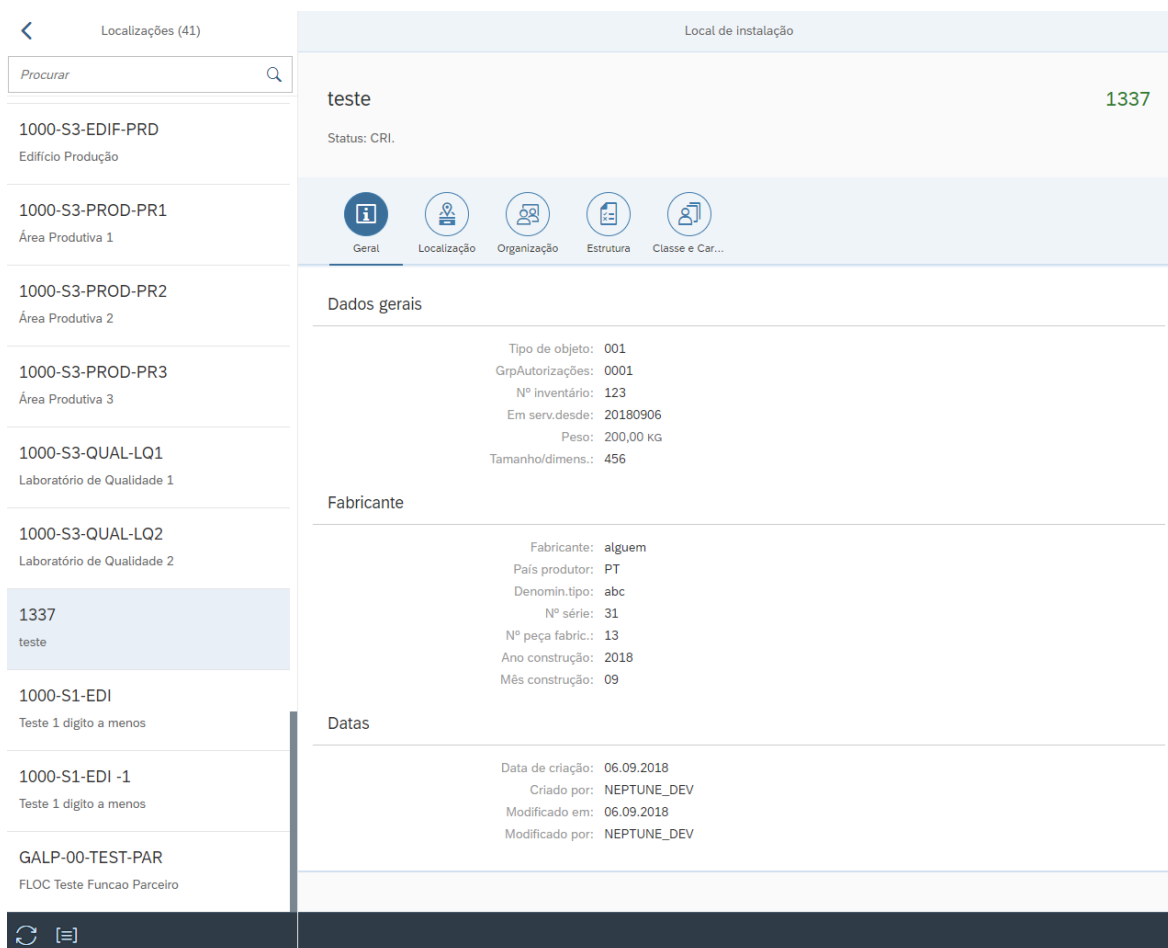


Figura 24 - Aplicação genérica de listagem de locais de instalação com detalhes.



### 5.2.2.3 Aplicação de ordens de trabalho só com detalhes

A aplicação de listagem e detalhes de ordens de trabalho, ilustrado na figura 25, apresentava as seguintes especificações:

1. O ecrã principal, deverá estar dividido em três partes.
  - a. À esquerda, em cima, campos de seleção para ajudar na pesquisa de ordens de trabalho;
  - b. À esquerda, por baixo dos campos de seleção, uma listagem de ordens de trabalho;
  - c. Do lado direito, depois de escolher uma ordem de trabalho, os seus detalhes.

The screenshot displays a web application interface for managing work orders. It is divided into two main sections: a left sidebar for search and a right pane for details.

**Left Sidebar (Search and List):**

- Search Bar:** Labeled "Procurar" with a magnifying glass icon.
- Filters:**
  - Orders Type/Date:** A dropdown menu currently showing "PM01 - Ordem de manutenção".
  - Only for my User:** A checked checkbox.
  - Created from:** A date picker showing "d 'de' MMM 'de' y".
  - Created to:** A date picker showing "d 'de' MMM 'de' y".
- Order List:** A table listing work orders. The first item is selected.
 

Date	Description	Order Number
29.03.2017	Manutenção Equipamento 1	4000000
29.03.2017	Manutenção Equipamento 1	4000001
31.03.2017	Manutenção Equipamento 2	4000002
28.08.2018		4000047
28.08.2018		4000048

**Right Pane (Order Detail):**

- Header:** "Order Detail" title. Content includes "LIB CNPA CAPC ERRD SCDM", "Manutenção Equipamento 1", and order number "4000000" with "PM01" as a sub-label.
- Activity Type:** "Inspeção".
- Information Icon:** A blue circle with a white 'i'.
- Head Data:**
  - Created by: João Rodrigues (00000010)
  - Planner Group - Plant: 1000
  - Main Work Center - Plant: PROD1 PRODUÇÃO 1 1000
  - Basic Start: 29.03.2017 00:00
  - Basic Finish: 29.03.2017 24:00
  - Loc. instalação: PROD-001-AA-00 Área de produção 1
  - Equipamento: 10000000 Equipment\_01\_TST
- Operations:** A table showing the work order's activities.
 

Activity	Work Center	Control Key	Description
10	PROD1	PM01	Manutenção Equipamento 1
- Partners:** A section with a table header (Function, Partner, Name, Address) and the text "Sem dados" (No data).

Figura 25 - Aplicação genérica de listagem de ordens de trabalho com detalhes, mas sem funcionalidades de PM.

A aplicação deve seguir o diagrama de casos de uso como na figura 26.

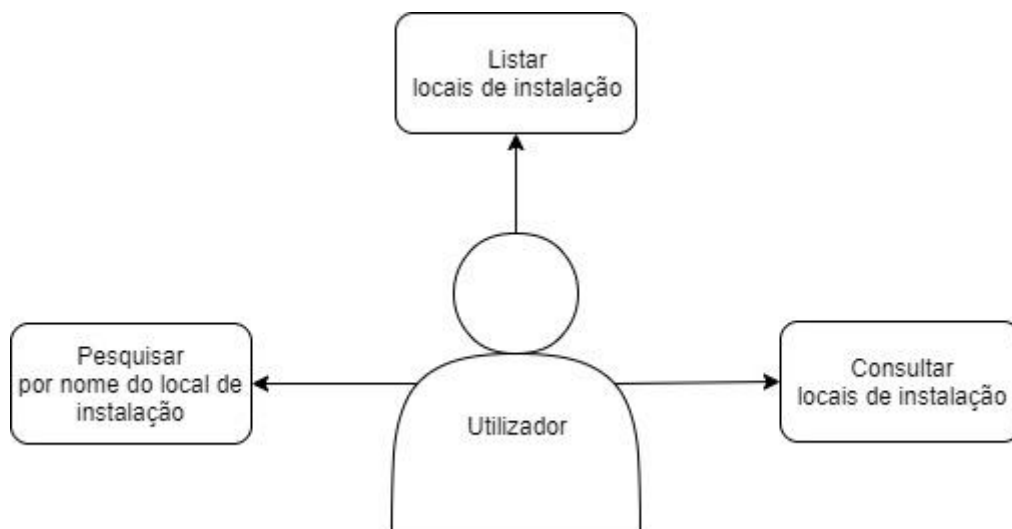


Figura 26 - Diagrama de casos de uso para aplicação de listagem de ordens de trabalho com detalhes, mas sem funcionalidades de PM.

#### 5.2.2.4 Aplicação de notificações com funcionalidades de PM

A aplicação de listagem e detalhes de notificações com funcionalidade de PM, apresentava as seguintes especificações, como ilustrado nas figuras 27 e 28:

1. Ao abrir a aplicação, o ecrã principal, deverá estar dividido em duas partes (Fig. 27).
  - a. Do lado esquerdo, campos de seleção para ajudar na pesquisa de notificações;
  - b. Do lado direito, uma listagem de notificações.
2. Por baixo da listagem de notificações, um botão '+' (mais) para criar uma nova notificação. Ao criar uma nova notificação, será aberto uma nova aplicação do tipo CRUD (Fig.28), de forma a que só se possa criar.
3. Ao selecionar uma notificação da listagem irá abrir a aplicação do tipo CRUD, mas em modo exibir para o documento de notificação. Ao abrir, todos os campos já estarão preenchidos com os dados da base de dados e bloqueados, de forma a que só possam ser visualizados.

4. O utilizador pode passar ao modo editar carregando no botão com o “lápiz” que se encontra no rodapé da aplicação. Ao carregar no botão, iram ser desbloqueados os campos para que possam ser editados. O “lápiz” será trocado por uns “oculos” para passar a modo exibir bloqueando todos os campos. As aplicações, por defeito, são abertas em modo de exibir.
5. Sempre que for para preencher campos com valores, o utilizador terá de usar uma ajuda de pesquisa para não cometer erros a preencher os campos, evitando validações.
6. A aplicação deverá ter 7 separadores:
  - a. Dados mestre
  - b. Dados de localização
  - c. Datas de Avaria e Parada
  - d. Sintomas
  - e. Pontos de medição
  - f. Comentários
  - g. Anexos
7. No canto inferior direito do rodapé, deverá estar um botão com o símbolo de uma disquete que serve para gravar o documento.
8. No centro do rodapé, deve de haver um botão para criar uma ordem de trabalho ligada diretamente à notificação aberta.
9. No separador de dados mestre tem de estar presente o tipo de notificação, descrição, local de instalação, número de equipamento, prioridade, número de utilizador que criou a notificação, início e fim obrigatório para resolver o problema, e numero da ordem de trabalho se tiver associada.
10. No separador de dados de localização, deve existir o centro de trabalho responsável, o centro onde está o problema, o grupo de planeamento, a localização, a sala, e o centro de custo a que é para serem aplicados os custos da manutenção.
11. No separador de datas de avaria e estado de paragem, tem de ter informações sobre o estado do problema. Se o equipamento está parado, quando é que começou a avaria. Também irá ter um campo para indicar a data do fim da avaria.



Notificação

!

Avaria Equipamento Produtivo A

10000050

Z2

Tipo

Status: MSPN

Criada: 03.08.2018

i

0

Tipo: Z2

Descrição:

Avaria Equipamento Produtivo A

Status usuário:

Loc. instalação:

1000-S1-PROD-PR1 Área Produtiva 1

Nº equipamento:

10000002 Equipamento PR-A

Prioridade:

Criado por:

0101427

Início obrigatório:

03.08.2018

11:14

Fim obrigatório:

00:00

Ordem de trabalho:

+

Create WO

Figura 28 - Aplicação genérica de CRUD de uma notificação em modo editar.

A aplicação deve seguir o diagrama de casos de uso como na figura 29.

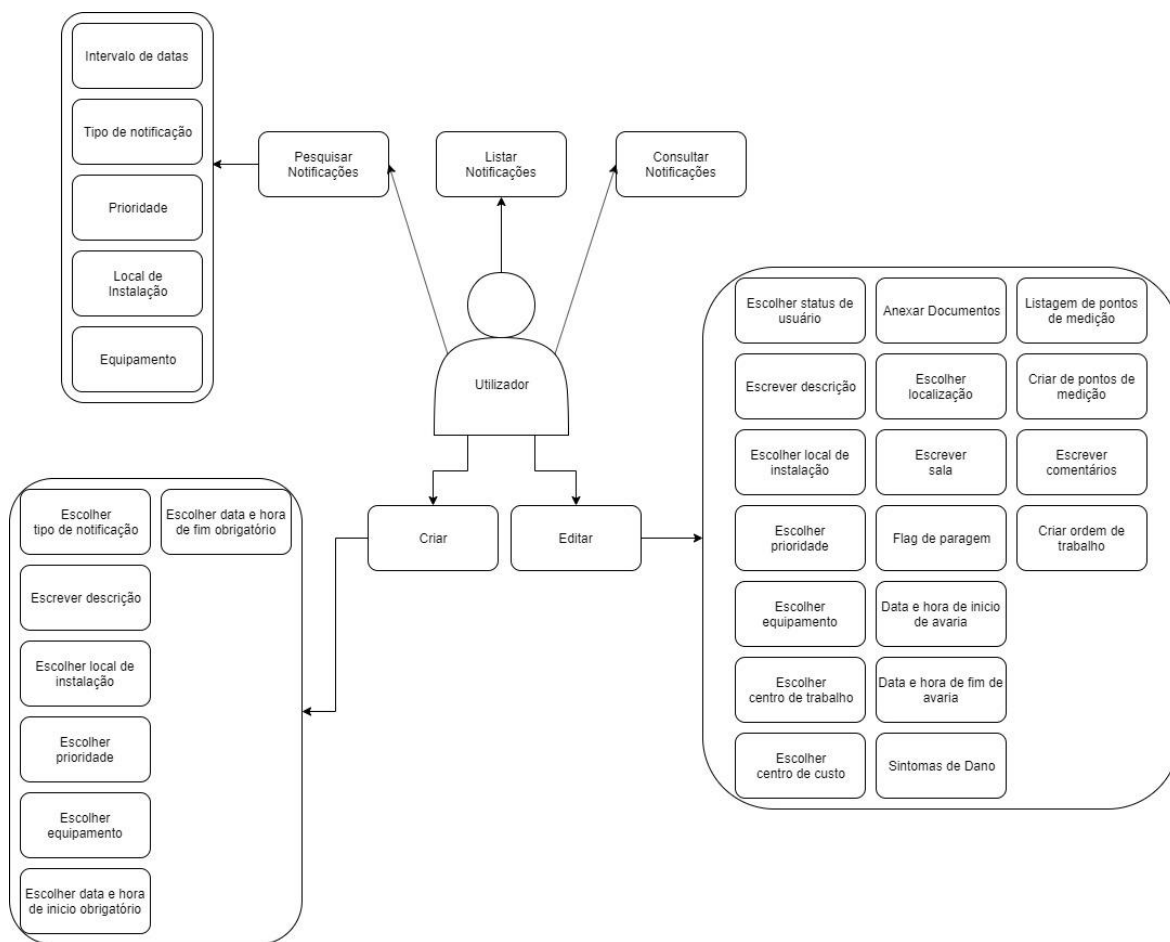


Figura 29 - Diagrama de casos de uso para aplicação de notificações genérica com funcionalidades de PM.

### 5.2.2.5 Aplicação de ordens de trabalho com funcionalidades de PM

A aplicação de listagem e detalhes de ordens de trabalho com funcionalidades de PM apresentava a mesma organização de design e funcionamento, apenas com algumas diferenças (Fig. 30).

1. Sempre que for para preencher campos com valores, o utilizador terá de usar uma ajuda de pesquisa para não cometer erros a preencher os campos, evitando validações.
2. A aplicação CRUD deverá ter 6 separadores (Fig. 31):
  - a. Dados mestre
  - b. Sintomas
  - c. Operações
  - d. Componentes
  - e. Comentários
  - f. Anexos
3. No canto inferior direito do rodapé, deverá estar um botão com o símbolo de uma disquete que serve para gravar o documento. Também deve de ter um botão para liberar a ordem e outro para fazer as confirmações das operações e componentes.
4. No separador de dados mestre tem de estar presente o tipo de ordem, descrição, *status* de sistema, *status* de utilizador, local de instalação, número de equipamento, prioridade, centro de localização, centro de custo, tipo de atividade e número da notificação se tiver associada.
5. No separador de sintomas, tem de apresentar campos para vários tipos de sintomas que o problema possa ter.
6. No separador de operações, tem de estar listado as operações a realizar e tem de haver funcionalidade CRUD (Fig. 32). Neste separador, para criar novas operações, deverá de abrir uma aplicação específica para essa tarefa.
7. No separador de componentes, tem de estar listado as componentes necessárias para realizar as operações e tem de haver funcionalidade CRUD. Neste separador, para criar novas componentes, deverá de abrir uma aplicação específica para essa tarefa.

- <
Seleção
Lista de resultados

---

▼ Data

Criado Desde:

Criado Até:

---

▼ Objeto

Tipo:

Fase:

Cen.Localiz:

Prioridade:

Loc.instalação:

Nº equipamento:

---

### Ordem de trabalho

Visão geral


**221** Entradas

Documento	Fase	Status
4000284	<span style="color: orange;">Released</span> 05.08.2019	LIB CAPC CCOP SCDM >
4000283	Created 05.08.2019	BLOQ ABER CCOP DMNV SCDM >
4000282 Teste	Created 26.07.2019	ABER CAPC SCDM >
4000281 Plano Manutenção Calendário	Created 25.07.2019	ABER CAPC DMNV SCDM >
4000280 Plano Manutenção Atividade Equip PR-B	<span style="color: orange;">Released</span> 24.07.2019	LIB CAPC DMNV NOLQ >
4000264 Teste Esquema Funcoes Parceiro	Created 12.07.2019	ABER CAPC SCDM >
4000263	<span style="color: orange;">Released</span> 11.07.2019	LIB CAPC NOLQ SCDM >
4000262	<span style="color: orange;">Released</span> 11.07.2019	LIB CAPC NOLQ SCDM >
4000261 Ruido estranho analisar	Completed 11.06.2019	TENC ENTE CONF CAPC MOME MatC NOLQ >
4000260	Created 31.05.2019	ABER CAPC DMNV >
4000259 ZPM2 - Notif - 10000343	Created 30.05.2019	ABER CAPC DMNV SCDM >
4000258 dffh	Created 30.05.2019	ABER CAPC DMNV ERRD SCDM >

44









Ordem de trabalho

 **4000280** ZPM2 TIPO

Plano Manutenção Atividade Equip PR-B

Criada: 24.07.2019 Fase: **Released**

   <sup>2</sup>  <sup>1</sup>   <sup>0</sup>

Tipo de ordem:	ZPM2
Status do sistema:	LIB CAPC DMNV NOLQ
Status usuário:	APR
Notificação:	
Descrição:	Plano Manutenção Atividade Equip PR-B
Prioridade:	
Centro de Localização:	1000
Local de Instalação:	1000-S1-PROD-PR3
Equipamento:	10000004 Equipamento PR-C
Centro de trabalho:	ELETRICA
Centro custo:	1199
Tipo ativid.PM:	002
Revisão:	
ReqC:	




  

Figura 31 - Aplicação genérica de CRUD de uma ordem de trabalho em modo editar.

Ordem de trabalho

4000280  
Plano Manutenção Atividade Equip PR-B  
Criada: 24.07.2019

ZPM2 TIPO  
Fase: Released

*i*

2

1

0

+

Oper	SOper	Descrição	Trabalho	Duração	Centro de Trabalho	Equipamento
0010		Operação 1	2 H	2 H	MECANICA	>
0020		Operação 2	1 H	1 H	MECANICA	>

Figura 32 - Aplicação genérica de CRUD de uma ordem de trabalho em modo editar listando operações de trabalho.

Confirmação Completa de Conclusão

**4000280**  
LIB CAPC DMNV NOLQ  
Criado: 24.07.2019

**ZPM2** TIPO  
Fase: **Released**

Sem dados

2

1

Confirmação do tempo

Confirmação	Operação	Descrição	Tipo de atividade	Trabalho atual	
801	0010	Operação 1	MEC_HN	0 H	>
802	0020	Operação 2	MEC_HN	0 H	>

Confirmar Tudo

Figura 33 - Aplicação genérica de confirmações de operações e componentes.

A aplicação deve seguir o diagrama de casos de uso como na figura 34.

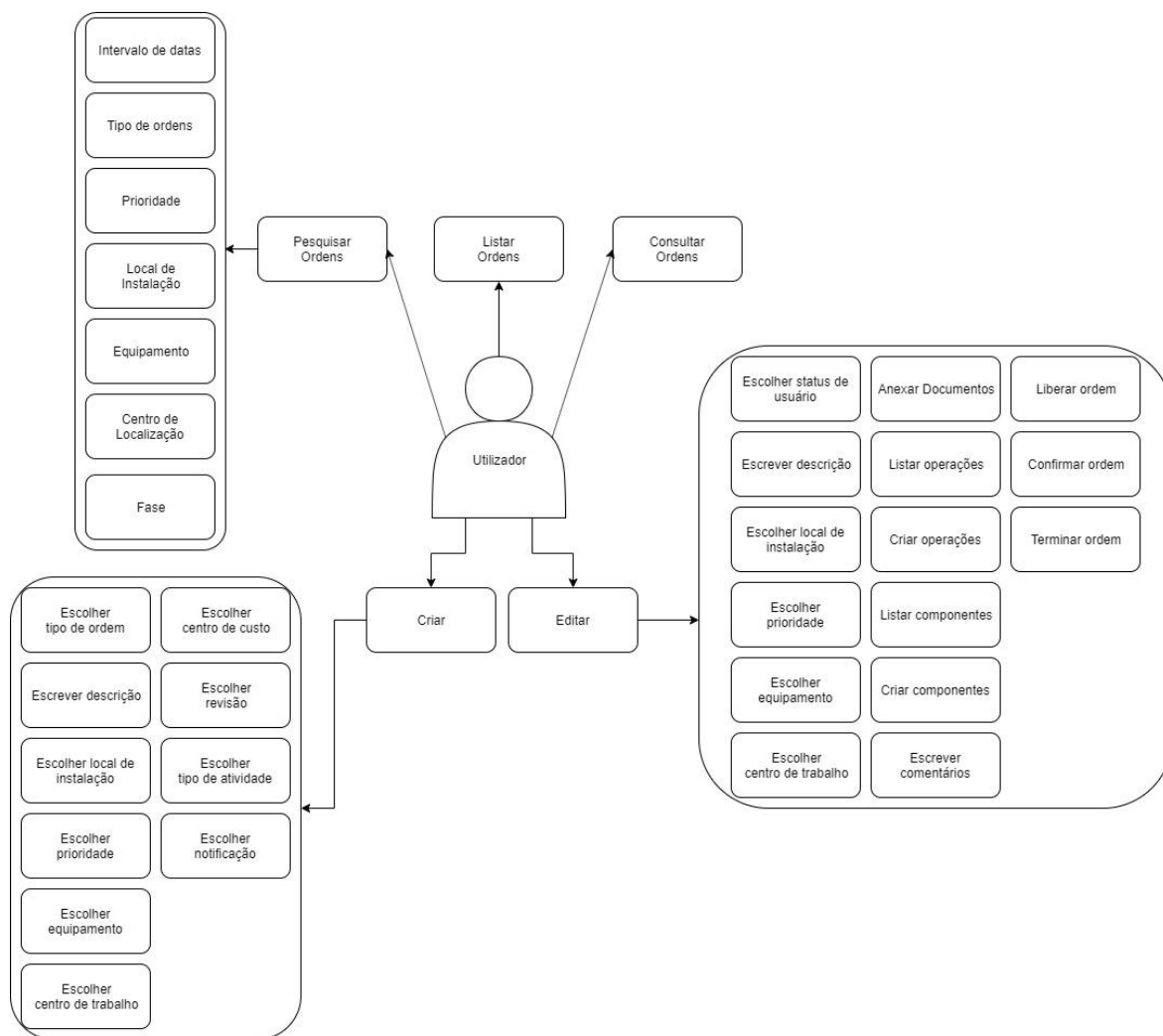


Figura 34 - Diagrama de casos de uso para aplicação genérica de ordens de trabalho com funcionalidades de PM.

### 5.2.3 Análise de Requisitos para aplicações vendidas a um cliente

As aplicações vendidas ao cliente A, foram (Fig. 35):

- Listagem e detalhes de equipamentos.
- Listagem, CRUD e funcionalidades de PM para notificações.
- Listagem, CRUD e funcionalidades de PM para ordens de trabalho.

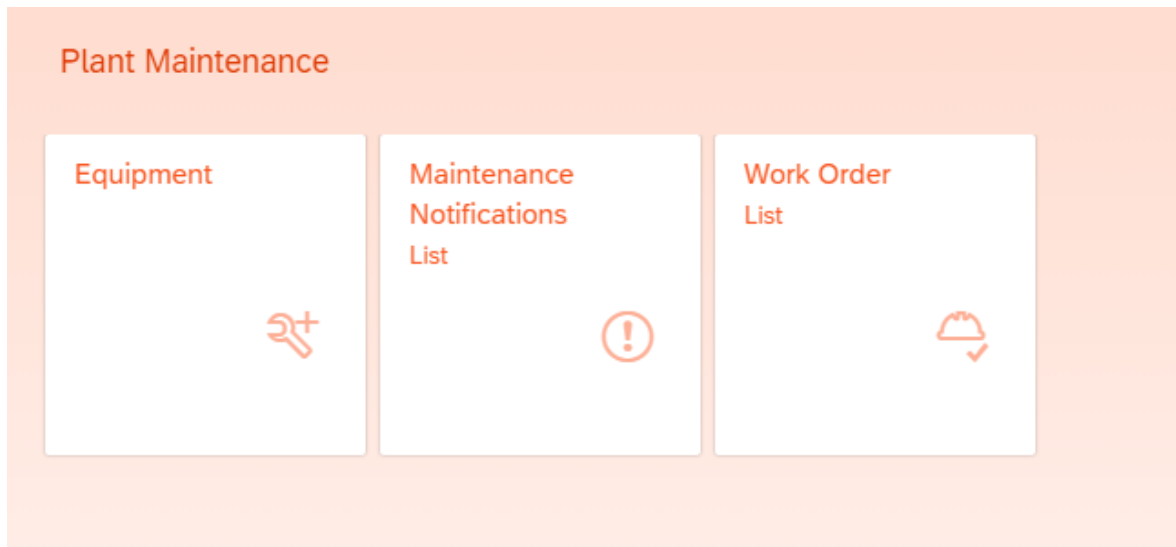


Figura 35 - Tiles das aplicações compradas pelo cliente no *launchpad* da sua máquina.

Apenas as aplicações de listagem com CRUD e funcionalidades de PM para notificações e ordens de trabalho precisaram de novos desenvolvimentos, para serem ajustadas e preparadas a funcionar com os requisitos do cliente.

Todas as aplicações principais e todas as aplicações dependentes, foram copiadas para a máquina do cliente de forma a que apenas essas fossem ajustadas com os novos requisitos.

A arquitetura de desenvolvimento para este cliente era diferente ao que estava a ser utilizado na *sandbox* da Softinsa. As aplicações deixaram de estar só numa máquina e passaram a estar em duas. Uma máquina aloja os desenvolvimentos em Neptune e outra máquina aloja os desenvolvimentos em ABAP. As máquinas comunicam entre si por uma ligação RFC.

### 5.2.3.1 Aplicação de notificações com funcionalidades PM

Os requisitos adicionais foram:

1. Acrescentar o campo Tag Nr. aos campos de seleção da aplicação de listagens e remover o local de instalação e equipamento (Fig. 36).
2. Acrescentar o campo Centro de Custo à aplicação de CRUD, que deverá vir preenchido com o valor que consta no dado mestre do equipamento, objeto de cabeçalho da nota, ou local de instalação caso não exista equipamento (Fig. 37).

Selecção

▼ Data

Criado desde:

d 'de' MMM 'de' y

Criado até:

d 'de' MMM 'de' y

▼ Objecto

Tipo:

Prioridade:

Tag Nr:

Limpar

Executar

+

Lista de resultados

!

Notificação

Visão Geral

107 Entradas

Nota	Criado	Status
130018931 M3 - cont 2	07.08.2019 U14129341	MSPR ORDA
130018930 M3 - cont 1	07.08.2019 U14129341	MSPR ORDA
130018920 M3 (ON) USTS (bad) QWE	06.06.2019 U14129341	MSPN
120265945 M2 - Simples A1	09.08.2019 U14129341	MSPN
120265936 M2 - Equip Off (2)	27.06.2019 U14129341	MSPN
120265935 M2 - Min Nep Off 2 (good)	30.05.2019 U14129341	MEAB MSPN
120265934 M2 - Min Nep Off	29.05.2019 U14129341	MEAB MSPR ORDA
120265933 M2 - Min Nep	16.04.2019 U14129341	Marcação para eliminação MREL MSEN MSPR ORDA
120265932 M2 - tst1	14.03.2019 U14129341	MSPR ORDA
120265931 Nota teste status usuario JFS01	14.01.2019 U10774823	MSPN

Figura 36 - Aplicação de listagem de documentos de notificações com funcionalidades de PM adaptada ao cliente A.

50

<

**M2 - Min Nep Off**  
120265934  
Status: MEAB MSPR ORDA

M2 Tipo  
Criada: 29.05.2019

CenTrab respon.: E-MEC

Grp.plnj.PM: ENF ENERFUEL

Localização:

Sala:

Centro custo: PT89-30003

Figura 37 - Aplicação de CRUD de uma notificação em modo editar.

A aplicação deve seguir o diagrama de casos de uso como na figura 38.

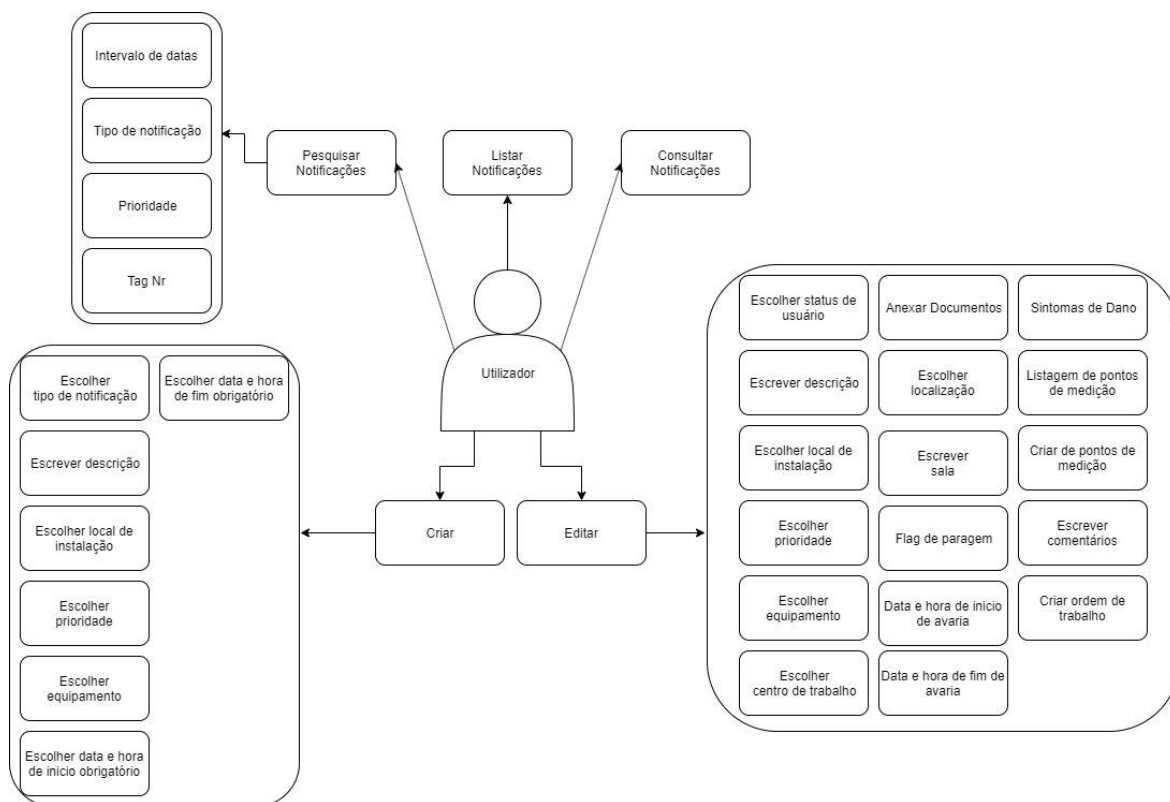


Figura 38 - Diagrama de casos de uso para aplicação para cliente de notificações com funcionalidades de PM.



### **5.2.3.2 Aplicação de ordens de trabalho com funcionalidades PM**

O requisito adicional à aplicação de listagem (Fig. 39) foi:

1. O campo Tag Nr. aos campos de seleção da aplicação de listagens e remover o centro, local de instalação e equipamento.

Para a aplicação do documento da ordem com CRUD e funcionalidades PM (Fig. 40), foi pedido:

1. Acrescentar o campo Centro de Custo, que deverá vir preenchido com o valor que consta no dado mestre do equipamento, objeto de cabeçalho da nota, ou local de instalação, caso não exista equipamento.
2. Acrescentar o campo Tipo de Atividade de Manutenção.
3. Acrescentar o campo Revisão.
4. Acrescentar o campo Requisição de Compra que será preenchido com a eventual requisição criada pela ordem na sua liberação.
5. Possibilitar a chamada de Listas de Tarefas para criar operações numa ordem.
6. Ter status de usuário ao nível das operações.
7. Possibilitar ver o stock de materiais ao selecionar os mesmos para uma ordem.

O requisito de possibilitar chamar listas de tarefas para criar operações numa ordem, não acabou por ser desenvolvido devido à sua complexidade e restrições de tempo para ser desenvolvido.

Selecção

✓ Data

Criado Desde:

d 'de' MMM 'de' y

Criado Até:

d 'de' MMM 'de' y

✓ Objeto

Tipo:

Fase:

Prioridade:

Tag Nr:

Lista de resultados

Ordem de trabalho

Visão geral

145 Entradas

Documento	Fase	Status	
22308893 M1 - tst 01	Criada 23.08.2019	ABER CAPC DMNV KKMP	>
22308892 TO22 - Mg1	Liberada 21.08.2019	LIB CAPC KKMP NOLQ	>
22308891 TO22 - Notif Off0	Criada 21.08.2019	ABER CAPC DMNV KKMP	>
22308890 TO22 - usts sync 2	Criada 21.08.2019	ABER CAPC DMNV ER RD	>
22308889 TO22 - usts sync 1	Criada 21.08.2019	ABER CAPC DMNV KKMP	>
22308888 TO22 - Comp 2	Criada 20.08.2019	ABER CAPC DMNV ER RD	>
22308887 TO22 - Comp 1	Criada 20.08.2019	ABER CAPC DMNV ER RD	>
22308886 TO22 - usts tst tbl	Criada 20.08.2019	ABER CAPC DMNV KKMP	>
22308885 TO22 - opr usts tst1	Criada 19.08.2019	ABER CAPC DMNV KKMP	>
22308884 TO22 - usts tst2	Criada 19.08.2019	ABER CAPC DMNV KKMP	>

Limpar

Executar

+

[=]

Figura 39 - Aplicação de listagem de documentos de ordens de trabalho com funcionalidades de PM adaptada ao cliente A.

54

<

M2 - Min Nep Off  
22308785

TO22 TIPO

Criada: 29.05.2019Fase: Criada

i

+<sup>1</sup>

+<sup>2</sup>

+<sup>1</sup>

+<sup>1</sup>

+<sup>0</sup>

Tipo de ordem: TO22

Status do sistema:

ABER CAPC DMNV

Status usuário:

APR

Notificação:

120265934

Descrição:

M2 - Min Nep Off

Prioridade:

Local de Instalação:

ENF-PRO-PTT

Equipamento:

20018453 Tricanter

Centro de trabalho:

E-MEC

Centro custo:

PT89-30003

Tipo atvid.PM:

04

Revisão:

ReqC.:

Figura 40 - Aplicação de CRUD de uma ordem de trabalho em modo editar.

A aplicação deve seguir o diagrama de casos de uso como na figura 41.

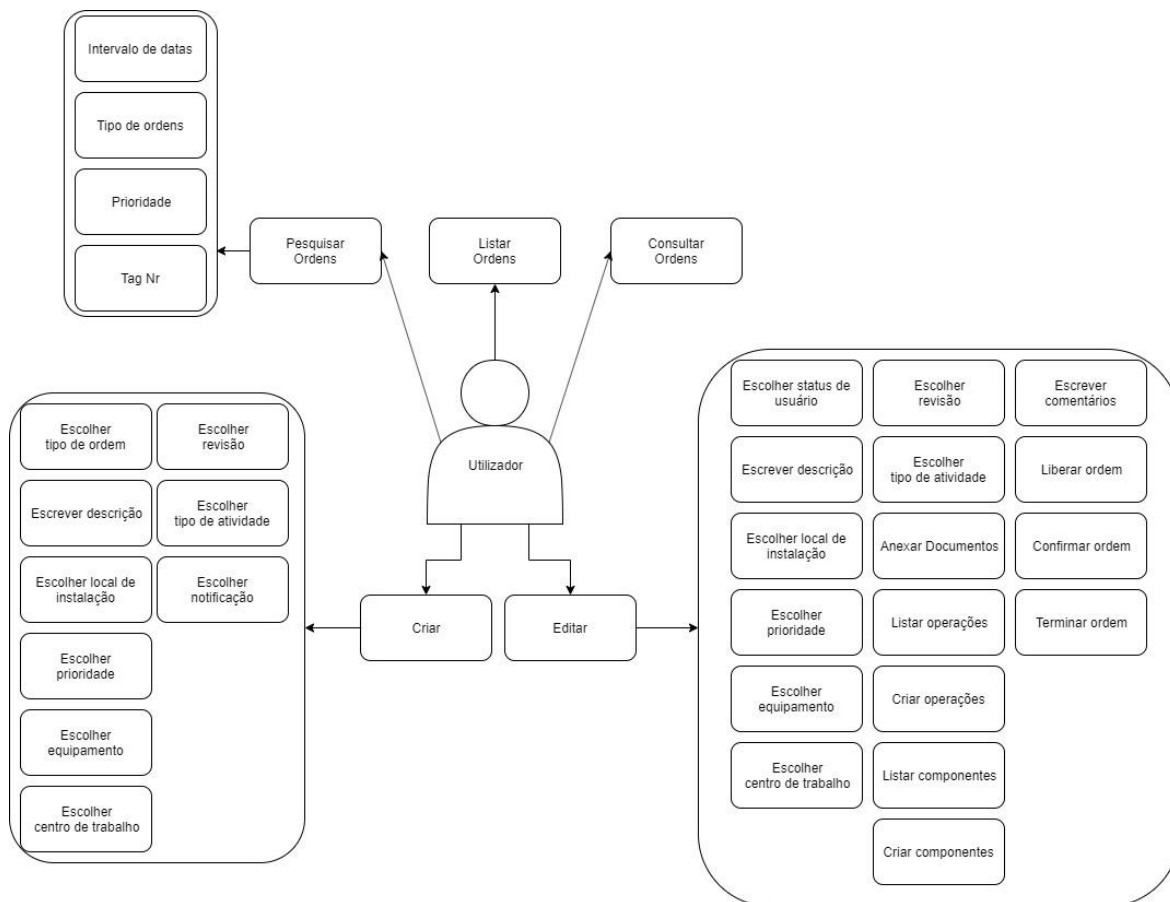


Figura 41 - Diagrama de casos de uso para aplicação para cliente de ordens de trabalho com funcionalidades de PM.

### 5.2.4 Transações Principais

Todas as aplicações do projeto, replicam o comportamento das transações de PM já existentes no ERP da SAP. Desta forma, foi necessário ao programados conhecer e saber funcionar com as principais transações de PM detalhadas da tabela 3, para aprender o seu comportamento e o conseguir replicar no desenvolvimento das aplicações.

Tabela 3 - Principais transações utilizadas

Transação	Funcionalidade
<b>IW21</b>	Criar Notificações
<b>IW22</b>	Editar Notificações
<b>IW23</b>	Exibir Notificações
<b>IW41</b>	Criar Ordens de Trabalho
<b>IW42</b>	Editar Ordens de Trabalho
<b>IW43</b>	Exibir Ordens de Trabalho
<b>IW44</b>	Confirmar Ordens de Trabalho
<b>IE01</b>	Criar Equipamento
<b>IE02</b>	Editar Equipamento
<b>IE03</b>	Exibir Equipamento
<b>IL01</b>	Criar Local de Instalação
<b>IL02</b>	Editar Local de Instalação
<b>IL03</b>	Exibir Local de Instalação

### 5.2.5 Conclusão

De acordo com as especificações fornecidas, o programador conseguiu com sucesso desenvolver todos os seus requisitos para as aplicações de demonstração e para as aplicações vendidas ao cliente A.

Utilizando a metodologia de trabalho SCRUM durante o projeto, fizeram-se várias reuniões com o colaborador funcional e verificou-se que era possível melhorar algumas especificações iniciais, sendo assim redefinidos alguns processos.

A adaptação das aplicações genéricas aos requisitos do cliente, levantou uma grande quantidade de questões e problemas para a qual as aplicações não estavam preparadas. Um desses problemas foi a grande quantidade de informação que era gerida do lado do cliente, o que demonstra que, as aplicações deveriam ser reajustadas com funcionalidades que não tinham sido especificadas previamente pelo cliente. Uma dessas funcionalidades, foi a implementação de uma limitação do volume de dados a que o utilizador tem acesso. Assim sendo, em reunião entre o programador, colaborador funcional e o chefe da equipa de ABAP, foi criada uma tabela de parametrização com apenas alguns dados específicos necessários ao projeto (Fig. 42). A inserção da tabela de parametrização teria de ser desenvolvida no início lógico das aplicações e só um projeto que utilizasse uma metodologia ágil seria capaz de responder com sucesso a tamanha modificação. Se a metodologia SCRUM não tivesse sido utilizada, estas alterações necessárias não seriam possíveis de aplicar, provando assim a capacidade do SCRUM de reagir de forma madura às principais mudanças do projeto. A tabela de parametrização iria dividir principalmente a empresa por armazéns (centros de trabalho) e assim, evitou-se desenvolver várias aplicações diferentes por armazém e apenas usar uma conforme parametrizada ao utilizador que a acede.

**Data Browser: Tabela ZPM\_NEP\_PARAMS 9 acertos**

MANDT	UNAME	WERKS	TPLNR	BEBER	IWERK	INGRP	ARBPL	QMART	AUART	WO_SYNC_FROM	WO_SYNC_TO
100						MGR		M1;M2;M3	TO22;TO28;TO29	20	30
100						MGR		M1;M2;M3	TO22;TO28;TO29	30	30
100						MGR		M1;M2;M3	TO22;TO28;TO29	30	30
100						ENF;MGR		M1;M2;M3	TO22;TO28;TO29	30	30
100						ENF;MGR		M1;M2;M3	TO22;TO28;TO29	30	30
100						ENF;MGR		M1;M2;M3	TO22;TO28;TO29	30	30
100			ADM*			ENF;MGR		M1;M2;M3	TO22;TO28;TO29	30	30
100						ENF;MGR		M1;M2;M3	TO22;TO28;TO29	30	30
100						ENF;MGR		M1;M2;M3	TO22;TO28;TO29	30	10

Figura 42 – Tabela de parametrização a utilizadores no cliente

Conclui-se também, que por vezes é melhor não começar a desenvolver aplicações a partir de aplicações de demonstração. Esta situação revelou-se por vezes complicada porque o programador teve de aprender e ajustar código já desenvolvido e desorganizado para que conseguisse realizar os seus desenvolvimentos.

Ao longo das especificações é possível acompanhar com as figuras os resultados finais das especificações.

## 5.3 Caso 2 | Implementação de Notas

### 5.3.1 Problema

A máquina *sandbox* da Softinsa necessitava que fosse implementada uma nota para corrigir uma anomalia no sistema.

A nota a ser implementada foi fornecido ao programador por um colaborador funcional da Softinsa.

Para realizar a implementação da nota, foi facultada as informações da máquina em que era necessário implementar a nota, dados de autenticação para o programador aceder à Máquina, bem como ao portal de suporte da SAP onde é possível analisar as notas e fazer o registo de chaves de objetos se necessário.

A metodologia utilizada para realizar esta tarefa foi a *Waterfall*, pois segue um conjunto de desenvolvimentos hierárquicos onde se tem de seguir a ordem estipulada. O não cumprimento da ordem de desenvolvimentos pode trazer problemas acrescidos ao sistema.

A implementação da nota necessária, iria corrigir dois sintomas de erros:

1. Na utilização da transação MEMASSRQ ou na BAPI\_PR\_CHANGE, quando era usada para fazer uma alteração na requisição de compra de serviços. Embora nenhum campo relacionado às linhas de serviço / contornos tenha sido alterado, o sistema valida campos inalterados que não são necessários.
2. Ao aceitar uma requisição de compra com a aplicação FIORI PR Approval, faz uma alteração no preço do serviço: os preços do serviço são divididos por 100 ou são definidos como zero 0, caso sejam usadas moedas sem decimais (por exemplo, HUF (Florim Húngaro), COP (Peso Colombiano), INR (Rupia Indiana), JPY (Iene)).

### 5.3.2 Análise de Requisitos

O programador acede à máquina e a primeira tarefa que faz é verificar que módulos estão instalados e em que versões se encontram como ilustrado na figura 43. De acordo com as especificações do cliente, o modulo que iria ser trabalhado era o de Logística, então o módulo analisado com mais detalhe foi o SAP\_APPL. É a partir da versão deste modulo que o programador tira as primeiras conclusões para poder realizar a tarefa de implementar a nota.



Component	Release	SP-Level	Support Package	Short Description of Component
SAP_APPL	617	0011	SAPKH61711	Logistics and Accounting
SAP_BASIS	740	0013	SAPKB74013	SAP Basis Component
SAP_BS_FND	747	0011	SAPK-74711INSAPBSFND	SAP Business Suite Foundation
SAP_BW	740	0013	SAPKW74013	SAP Business Warehouse
SAP_FIN	720	0004	SAPK-72004INSAPFIN	SAP_FIN
SAP_GWFND	740	0020	SAPK-74020INSAPGWFND	SAP Gateway Foundation
SAP_HR	604	0093	SAPKE60493	Human Resources

Figura 43 – Modulos, releases e support package instalados na sandbox da Softinsa.



A segunda tarefa que o programador faz, é analisar a nota em específico que é pretendida ser implementada e comparar a versão da *release* e do *support package* com o da máquina do cliente (Fig. 44).

Se a *release* e do *support package* da nota forem superiores ao da máquina, a nota é possível de ser implementada.

1857373 - Additional validation of service lines using MEMASSRQ or FIORI PR approval application <small>Version 6 from 28/08/2018 in English</small>						
Description	Software Components	Correction Instructions	Support Packages	This document is referenced by	Attributes	Languages
Support Packages						
Software Component Version			Support Package			
SAP_APPL 616			SAPKH61611			
SAP_APPL 617			SAPKH61714			

Figura 44 - Modulo, *release* e *support package* em que é para implementar a nota.

A terceira tarefa do programador, e muito importante, é verificar se não existem dependências da nota pretendida de ser implementada. Por vezes, existe notas que dependem da implementação de outras notas primeiro, para que o problema que se pretende corrigir, possa ser efetivamente corrigido.

Por fim, o programador deve de criar para seu auxílio, um documento de passos realizados durante a implementação da nota. Este documento será guardado depois da implementação para consulta caso seja necessário implementar em outro cliente a mesma nota e apareça alguma duvida durante a implementação da mesma.

### 5.3.3 Resolução

O programador abriu a nota pretendida de ser implementada no portal de suporte da SAP e fez uma análise à nota. Verificou que a nota não tinha passos manuais para realizar e também não tinha dependências de outras notas como ilustrado na figura 45.

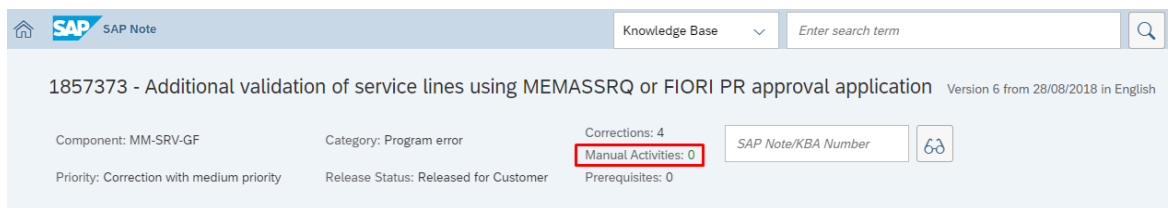


Figura 45 - Indicação de número de passoaas manuais da nota.

Esta nota, não tendo dependências de outras notas, pode ser implementada sem ser necessário fazer mais validações a outras notas que deveriam de ser implementadas primeiro.

Caso esta nota tivesse passos manuais para desenvolver, o programador teria de verificar primeiro se os passos se aplicavam ao sistema e também teria de verificar em que altura deveria de os desenvolver. Existem passos pré e pós implementação, que devem de ser desenvolvidos conforme especificado para corrigir o problema.

A partir deste momento, o programador começa a documentação de todos os passos para implementar a nota.

O programador começa por ir à transação SNOTE e fez o *download* da nota. Depois do *download* ter acabado, verificou novamente que a nota pode ser implementada na máquina, como demonstra a figura 46.

Note	Version	Short text	Component	Status	Implementation Stat.	User
1857373	6	Additional validation of service lines using MEMASSRQ or FIORI PR approval appli	MM-SRV-GF	new	Can be implemented	

Figura 46 - Detalhe da nota antes de ser implementada.

Assim sendo, começou a implementar a nota, verificando passo a passo todo o processo para garantir que nada de errado acontecesse durante a sua implementação.

Na figura 47, pode ser visto o objeto que será alterado com a implementação da nota e o seu estado.

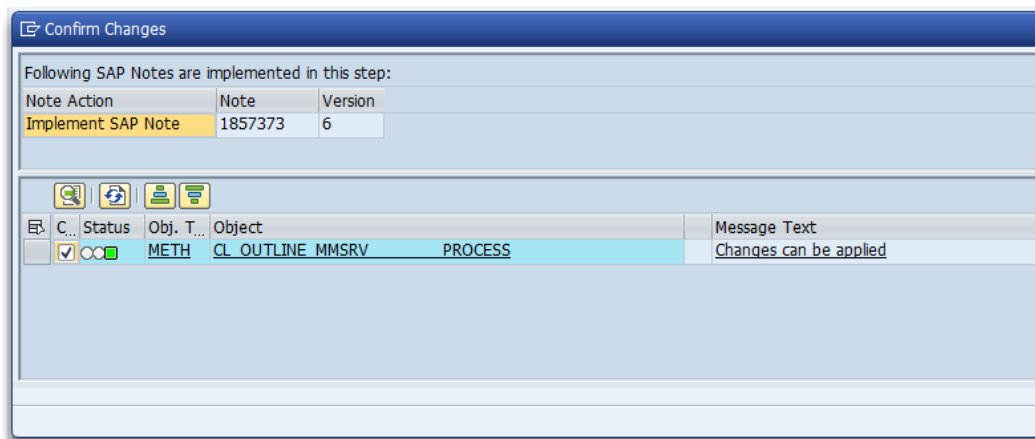


Figura 47 - Estado do objeto a ser alterado pela nota.

Como demonstra a figura 48, após serem realizadas as alterações, o objeto tem de ser ativado para que as mesmas alterações passem a ser usadas. Se não for ativado, a versão antes da correção é a que será usada.

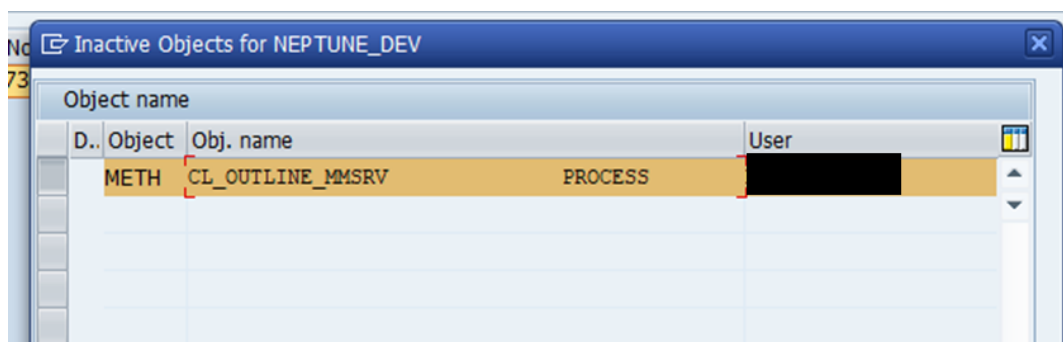


Figura 48 - Lista de objetos que têm de ser ativados depois da implementação da nota.

Por fim, surge a informação de que a nota foi completamente implementada (Fig. 49). Agora podem ser realizados testes para verificar se o problema para qual a nota foi implementada está corrigido. Se não estiver corrigido, será necessário implementar outra nota corretiva para um novo sintoma de erro que não foi analisado.

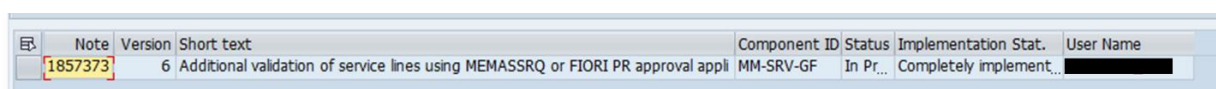


Figura 49 - Detalhe da nota depois de ser implementada.

### 5.3.4 Conclusão

Depois da implementação da nota, o colaborador funcional foi informado que as alterações foram realizadas. É facultada a lista de todas as notas implementadas e o registro de objetos que foi necessário efetuar.

Só utilizando a metodologia de trabalho *Waterfall* é que se consegue implementar notas, pois elas consistem na instalação de um conjunto organizado e estrito de tarefas para realizar correções de erros no sistema. Essas tarefas podem ser realizadas através dos passos manuais feitos antes ou depois de implementar a nota, ou apenas fazer a implementação da nota. Os passos manuais podem ser alterações do tipo de dados de campos de tabelas, inserir ou apagar campos de tabelas, alterações a textos de mensagens, entre outros. A implementação de uma nota, pode fazer correções a código de determinadas funções utilizadas no sistema ERP. Se essa ordem de tarefas não for seguida, a nota não conseguirá ficar implementada e irá criar inconsistências no sistema. Apesar da metodologia *Waterfall* ser considerada ultrapassada por não ser flexível, continua a ser necessária para estes processos, em que é necessário seguir um conjunto de tarefas sem sair fora do fluxo lógico.

Todas as notas são desenvolvidas pela SAP e disponibilizadas às empresas que desenvolvem soluções SAP para corrigir os seus sistemas.

O documento criado pelo programador com a informação passo a passo de como foi implementada a nota, é guardado num repositório para consulta futura.

## Capítulo 6 | Conclusões

Nos dias de hoje, os avanços tecnológicos têm um papel fulcral nas organizações, criando sistemas capazes de armazenar todos os dados de uma empresa, sendo possível disponibiliza-los em tempo real, facilitando a sua consulta, análise e recolha de informação, e desta forma, tem permitido e simplificado a tomada de decisão antecipada nas empresas e o planeamento eficaz a longo prazo.

Para descrever as principais conclusões do estágio, é importante mencionar a escolha pela modalidade estágio e os seus objetivos, pois sobre eles recaem todo o trabalho desenvolvido ao longo do mesmo.

O estágio curricular é uma experiência que permite estimular e reforçar as competências de um estudante, exercendo práticas devidamente qualificadas no âmbito da atividade profissional. Através desta modalidade, foi possível adquirir competências e conhecimentos práticos da realidade laboral, pois o estágio visa facultar uma experiência específica que facilite e promova um estudo aprofundado numa determinada área. Para além do crescimento de capacidades de trabalho, também permite crescer a nível pessoal, ganhando grandes responsabilidades profissionais, horários a cumprir e *deadlines* no tempo de execução de tarefas.

O objetivo geral deste estágio foi compreender o funcionamento de uma organização de topo na área das tecnologias de informação como a Softinsa. Através da realização deste estágio foram adquiridas competências a nível da programação, de forma a que seja possível a criação e manutenção tanto de aplicações do ERP da SAP como de aplicações Neptune. Foi dada a oportunidade de trabalhar na área de consultadoria SAP ABAP e Neptune, em que o objetivo principal foi desenvolver e apresentar aplicações de PM genéricas que acabaram por ser vendidas a um cliente da Softinsa. Para além do desenvolvimento de aplicações, também estive envolvido numa tarefa de manutenção da máquina *sandbox* da Softinsa, implementando uma nota corretiva que serviu para corrigir erros no sistema.

A implementação de notas é uma tarefa que requer uma análise não só da *release* e do *support package* do sistema, mas também dos sintomas dos erros que é suposto as notas corrigirem. O objetivo das notas é a correção de possíveis erros que o sistema possa conter, para melhorar os processos. É uma tarefa que tem de ser realizada com rigor e com cuidado, visto que qualquer falha no desenvolvimento dos passos manuais ou da implementação da nota pode trazer inconsistências ao sistema em funcionamento.

A criação de soluções SAP usando o *addon* Neptune foi o principal foco do estágio. Soluções SAP *mobile* ainda são novidade, mas já deram provas de que são muito vantajosas para as empresas. O uso desta plataforma de desenvolvimento acelerado permite a criação de aplicações com menores prazos e custos em comparação com o desenvolvimento de programas comuns, devido à sua arquitetura de desenvolvimento *drag and drop*. As funcionalidades base das aplicações genéricas que foram solicitadas inicialmente, são as necessárias para um projeto de PM funcionar eficazmente com os requisitos mínimos. Assim que as aplicações são vendidas a um cliente, é necessária que estas sejam adaptadas aos requisitos específicos do cliente. O uso de uma metodologia ágil no desenvolvimento deste tipo de projeto tem um papel muito importante nas várias fases de adaptação das aplicações às necessidades do cliente. As aplicações desenvolvidas vão ser utilizadas quer em dispositivos móveis, quer em plataformas do tipo *desktop*.

Uma vez que o *software addon* Neptune é recente, ainda existe grande margem para estudar e explorar esta nova tecnologia. É necessário acompanhar a evolução desta tecnologia, realizando um estudo constante da ferramenta e das suas funcionalidades. Durante o estágio tive a oportunidade de participar em palestras dadas pela Neptune Software sobre o *addon* Neptune, as suas capacidades e os futuros desenvolvimentos. Paralelamente, também é importante apostar na melhoria e utilização das linguagens de programação ABAP e Javascript, que são a base das aplicações em Neptune.

Visto que o desempenho das aplicações é uma das características chave do seu desenvolvimento, a otimização de código é um fator de extrema importância. Este trabalho dá a conhecer uma área das tecnologias que é desconhecida para alguns estudantes, devido à liberdade de escolhas e de decisões que se ganha nível académico. Permite não só mostrar os conhecimentos que são utilizados na área de SAP desde Base de Dados, Javascript entre outros, como também dá a conhecer algumas das metodologias que podem ser utilizadas no mercado de trabalho. É bastante importante conhecer os limites da tecnologia para se tirar o máximo desempenho e aproveitamento possível.





## **Referências Bibliográficas**

- [1] Campos, J. (Agosto de 2015). Pirâmide ideal de decisão pragmática. Consultado em Janeiro de 2019, de ResearchGate: [https://www.researchgate.net/figure/Figura-1-Piramide-ideal-da-decisao-pragmatica-Os-dados-existentis-para-a-gestao-de\\_fig1\\_281285772](https://www.researchgate.net/figure/Figura-1-Piramide-ideal-da-decisao-pragmatica-Os-dados-existentis-para-a-gestao-de_fig1_281285772)
- [2] Sap. (2019). SAP: A 47-year history of success. Consultado em Janeiro de 2019, de Sap: <https://www.sap.com/corporate/en/company/history.html>
- [3] Computerworld. (4 de Novembro de 2013). IBM inaugura Centro de Inovação Tecnológica de Tomar. Consultado em Outubro de 2019, de Computerworld: <https://www.computerworld.com.pt/2013/11/04/ibm-inaugura-centro-de-inovacao-tecnologica-de-tomar/>
- [4] Softinsa. (2019). Presidente da República inaugura novo Centro de Inovação Tecnológica de Viseu. Consultado em Outubro de 2019, de Softinsa: <https://www.softinsa.pt/noticia?Presidente-da-Repblica-inaugura-novo-Centro-de-Inovao-Tecnologica-de-Viseu>
- [5] Softinsa. (2019). SOBRE A SOFTINSA. Consultado em Outubro de 2019, de Softinsa: <https://www.softinsa.pt/softinsa>
- [6] Softinsa. (2019). SAP CONSULTING. Consultado em Outubro de 2019, de Softinsa: <https://www.softinsa.pt/sap>
- [7] Softinsa. (2019). CENTROS DE INOVAÇÃO TECNOLÓGICA. Consultado em Outubro de 2019, de Softinsa: <https://www.softinsa.pt/centros-de-inovacao>

[8] Roy, P. (12 de Setembro de 2018). What are GTS and GBS in IBM? Which is a better grade?. Consultado em Outubro de 2019, de Quora: <https://www.quora.com/What-are-GTS-and-GBS-in-IBM-Which-is-a-better-grade>

[9] Adviser IT. (25 de Janeiro de 2018). Conheça a função de um consultor SAP. Consultado em Outubro de 2019, de Adviser IT: <https://adviserit.com.br/conheca-funcao-de-um-consultor-sap/>

[10] JOBHERO. (2019) What is a sap consultant. Consultado em Dezembro de 2019, de JOBHERO: <https://www.jobhero.com/what-is-a-sap-consultant/>

[11] Prof Lucas, A., Mestre Pedron, C., Mestre Naves, F., Eng<sup>a</sup> Silva, F., Dr Camanho, J., Dr Henriques, L., Conceitos fundamentais de sistemas e tecnologias de informação e de gestão do conhecimento. Consultado em Outubro de 2019, de ISEG: <https://www.iseg.ulisboa.pt/aquila/getFile.do?fileId=8003&method=getFile>

[12] Diversidades & Vivências. (20 de Maio de 2015). Vantagens e desvantagens de um Sistema de Informação. Consultado em Outubro de 2019, de Diversidades & Vivências: <http://diversidadesevivencias.blogspot.com/2015/05/vantagens-e-desvantagens-de-um-sistema.html>

[13] Batista, R (1 de Junho de 2014). Sistemas de informação: um estudo comprovativo das vantagens e desvantagens do uso de softwares integrados e não integrados. Consultado em Dezembro de 2019, de Slideshare: <https://pt.slideshare.net/rafaelarthurbatista/sistemas-de-informaes-um-estudo-comparativo-das-vantagens-e-desvantagens-do-uso-de-softwares-integrados-e-no-integrados>

[14] Santos, L (Março 2016). A adoção de sistemas de informação na gestão estratégica de uma organização. Consultado em Dezembro de 2019, de Repositório Científico IPVC: [http://repositorio.ipvc.pt/bitstream/20.500.11960/1622/1/Luis\\_Santos.pdf](http://repositorio.ipvc.pt/bitstream/20.500.11960/1622/1/Luis_Santos.pdf)

[15] Guru99. (2019). What is SAP? Definition of SAP ERP Software. Consultado em Janeiro de 2019, de Guru99: <https://www.guru99.com/what-is-sap-definition-of-sap-erp-software.html>

[16] IBM. (Dezembro de 2019). SAP. Consultado em Dezembro de 2019, de IBM: <https://www.ibm.com/uk-en/services/sap>

[17] SAP. (2019) Enterprise management ERP. Consultado em Dezembro de 2019, de SAP: <https://www.sap.com/portugal/products/enterprise-management-erp.html>

[18] Bispo, M. (5 de Agosto de 2019). Apostila de SAP para iniciantes, Manuais, Projetos, Pesquisas de Programação em Windows. Consultado em Outubro de 2019, de Docsity: <https://www.docsity.com/pt/apostila-de-sap-para-iniciantes/4934482/>

[19] Tutorialspoint. (2019). Sap Modules. Consultado em Dezembro de 2019, de Tutorialspoint: [https://www.tutorialspoint.com/sap/sap\\_modules.htm](https://www.tutorialspoint.com/sap/sap_modules.htm)

[20] Microsoft Dynamics 365. (2019). O que é o ERP e porque precisa dele?. Consultado em Janeiro de 2019, de Microsoft Dynamics 365: <https://dynamics.microsoft.com/pt-pt/erp/what-is-erp/>

[21] Marquez, G. (2017). Vantagens e Desvantagens ERP. Consultado em Janeiro de 2019, de NFE: <https://nfe.io/blog/gestao-empresarial/vantagens-e-desvantagens-erp/>

[22] Zadeh, A. Akinyemi, B. Jeyaraj, A. Zolbain, H. (Setembro de 2018). Cloud ERP Systems for Small-and-Medium Enterprises: A Case Study in the Food Industry. Consultado em Dezembro de 2019, de ResearchGate: [https://www.researchgate.net/publication/327370130\\_Cloud\\_ERP\\_Systems\\_for\\_Small-and-Medium\\_Enterprises\\_A\\_Case\\_Study\\_in\\_the\\_Food\\_Industry](https://www.researchgate.net/publication/327370130_Cloud_ERP_Systems_for_Small-and-Medium_Enterprises_A_Case_Study_in_the_Food_Industry)

[23] Linton, I. (2019). Pros & Cons of ERP Systems for Small Businesses. Consultado em Dezembro de 2019, de Chron Small Business: <https://smallbusiness.chron.com/pros-cons-erp-systems-small-businesses-40555.html>

[24] Tutorials Point. (2019). Sap ABAP Overview. Consultado em Janeiro de 2019, de Tutorials Point: [https://www.tutorialspoint.com/sap\\_abap/sap\\_abap\\_overview.htm](https://www.tutorialspoint.com/sap_abap/sap_abap_overview.htm)

[25] Tutorials Point. (2019). Sap Fiori Introduction. Consultado em Janeiro de 2019, de Tutorials Point: [https://www.tutorialspoint.com/sap\\_fiori/sap\\_fiori\\_introduction.htm](https://www.tutorialspoint.com/sap_fiori/sap_fiori_introduction.htm)

[26] Sap. (2019). Design. Develop. Deliver. A modern UX for every device. And every user. Consultado em Janeiro de 2019, de Sap: <https://www.sap.com/products/fiori.html>

[27] Neptune Software. (2019). About Neptune Software and what it does for you. Consultado em Janeiro de 2019, de Neptune Software: <https://www.neptune-software.com/about/>

[28] Priyanka, J. (2018). DX Platform. Consultado em Janeiro de 2019, de Neptune Software: <https://community.neptune-software.com/p/dx-platform>

[29] Neptune Software. (15 de Fevereiro de 2017). Neptune Software UX Platform - Overview. Consultado em Fevereiro de 2019, de Youtube: <https://youtu.be/5dceJkGqQ4w>

[30] Hager, C. (2019). Neptune Software – SAP Fiori Apps in only a few days!. Consultado em Janeiro de 2019, de Retail Solutions: <https://www.retailsolutions.ch/en/portfolio/solutions/neptune-software.html>

[31] Jain, P. (2018) Why should I invest in Neptune when Fiori is Freeori?. Consultado em Dezembro de 2019, de Neptune Software Community: <https://community.neptune-software.com/documentation/why-should-i-invest-in-neptune-when-fiori-is-freeo>

[32] SOA People. (26 de Abril de 2018). 6 reasons why sap leonardo is the future of intelligent erp. Consultado em Janeiro de 2019, de SOA People: <https://www.soapeople.com/blog/6-reasons-why-sap-leonardo-is-the-future-of-intelligent-erp>

[33] Sap. (2019). Unlock the Intelligent Enterprise with SAP Leonardo Intelligent Technologies. Consultado em Janeiro de 2019, de Sap: <https://www.sap.com/products/leonardo.html>

[34] Sap. (2019). Internet of Things (IOT). Consultado em Dezembro de 2019, de Sap: <https://www.sap.com/products/leonardo/iot.html?infl=914c9020-1e31-46c5-83df-9dab52412ccc>

[35] Techopedia. (2017). Sap HANA. Consultado em Janeiro de 2019, de Techopedia: <https://www.techopedia.com/definition/28540/sap-hana>

[36] Rouse, M. (Junho de 2017). Sap HANA. Consultado em Janeiro de 2019, de SearchSAP: <https://searchsap.techtarget.com/definition/HANA-SAP-HANA>

[37] Mdsaptech. (2019). Sap HANA. Consultado em Janeiro de 2019, de Mdsaptech: <http://www.mdsaptech.com/en/MDSap-Solutions/12/16/Data-Warehousing/SAP-HANA>

[38] Sap. (2019). Hana. Consultado em Dezembro de 2019, de Sap: <https://www.sap.com/products/hana.html>

[39] Junior, C. (31 de Maio de 2017). Conheça e entenda as 12 principais metodologias de gestão de projetos. Consultado em Outubro de 2019, de Project Builder: <https://www.projectbuilder.com.br/blog/metodologias-de-gestao-de-projetos/>

[40] Capenda, A. (Outubro de 2015). Metodologia em Gestão Informática. Consultado em Dezembro de 2019, de Repositório Universidade Nova: <https://run.unl.pt/bitstream/10362/17374/1/TGI0050.pdf>

[41] Soares, J. (Outubro de 2016). Metodologias Ágeis na Gestão de Projetos e Não Tecnológicos – Caso de Estudo. Consultado em Dezembro de 2019, de Repositório Universidade de Lisboa: <https://www.repository.utl.pt/bitstream/10400.5/12956/1/DM-JMCS-2016.pdf>

[42] Brasileiro, R. (2017). Métodos Ágeis: O que é e porque você deve saber o que é. Consultado em Outubro de 2019, de Método Ágil: <http://www.metodoagil.com/metodos-ageis/>

[43] State of Agile. (2019). Introducing the 13th Annual State of Agile Report!. Consultado em Outubro de 2019, de State of Agile: <https://www.stateofagile.com>

[44] Desenvolvimento Ágil. (2014). SCRUM. Consultado em Outubro de 2019, de Desenvolvimento Ágil: <https://www.desenvolvimentoagil.com.br/scrum/>

[45] Vieira, D. (26 de junho de 2014). Scrum: A Metodologia Ágil Explicada de forma Definitiva. Consultado em Outubro de 2019, de MindMaster Educação Profissional: <http://www.mindmaster.com.br/scrum/>

[46] Scrum. (2019). Scrum. Consultado em Dezembro de 2019, de The Home of Scrum: <https://www.scrum.org/>

[47] Cole, A. (08 de Setembro de 2010). MODELO CASCATA PARTE I. Consultado em Outubro de 2019, de Qualidade em foco: <https://anielacole.wordpress.com/2010/09/08/modelo-cascata/>

[48] Moura, D. (27 de Junho de 2016). Abordagem Tradicional x Abordagem Ágil para Desenvolvimento de Software. Consultado em Outubro de 2019, de Linkdin: <https://pt.linkedin.com/pulse/abordagem-tradicional-x-%C3%A1gil-para-desenvolvimento-de-software-moura>

[49] Smartsheet. (2019). Waterfall. Consultado em Dezembro de 2019, de Smartsheet: <https://www.smartsheet.com/content-center/best-practices/project-management/project-management-guide/waterfall-methodology>

[50] Eder, S. Conforto, E. Amaral, D. Silva, S. (18 de Abril de 2014). Diferenciando as abordagens tradicional e ágil de gerenciamento de projetos. Consultado em Dezembro de 2019, de Scielo: <https://dx.doi.org/10.1590/S0103-65132014005000021>